



University of **HUDDERSFIELD**

University of Huddersfield Repository

Wolinski, Paul

Decomposition Theory: A Practice-Based Study of Popular Music Composition Strategies

Original Citation

Wolinski, Paul (2019) Decomposition Theory: A Practice-Based Study of Popular Music Composition Strategies. Doctoral thesis, University of Huddersfield.

This version is available at <http://eprints.hud.ac.uk/id/eprint/35278/>

The University Repository is a digital collection of the research output of the University, available on Open Access. Copyright and Moral Rights for the items on this site are retained by the individual author and/or other copyright owners. Users may access full items free of charge; copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational or not-for-profit purposes without prior permission or charge, provided:

- The authors, title and full bibliographic details is credited in any copy;
- A hyperlink and/or URL is included for the original metadata page; and
- The content is not changed in any way.

For more information, including our policy and submission procedure, please contact the Repository Team at: E.mailbox@hud.ac.uk.

<http://eprints.hud.ac.uk/>

Decomposition Theory

A practice-based study of popular music composition strategies.

Paul Wolinski

A commentary accompanying the creative portfolio submitted to the University of Huddersfield in partial
fulfilment of the requirements for the degree of Doctor of Philosophy.

October 2019

© University of Huddersfield 2019

Copyright Statement

The following notes on copyright and the ownership of intellectual property rights must be included as written below:

- i. The author of this thesis (including any appendices and/ or schedules to this thesis) owns any copyright in it (the “Copyright”) and they have given The University of Huddersfield the right to use such Copyright for any administrative, promotional, educational and/or teaching purposes.
- ii. Copies of this thesis, either in full or in extracts, may be made only in accordance with the regulations of the University Library. Details of these regulations may be obtained from the Librarian. Details of these regulations may be obtained from the Librarian. This page must form part of any such copies made.
- iii. The ownership of any patents, designs, trademarks and any and all other intellectual property rights except for the Copyright (the “Intellectual Property Rights”) and any reproductions of copyright works, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property Rights and Reproductions cannot and must not be made available for use without permission of the owner(s) of the relevant Intellectual Property Rights and/or Reproductions.

Contents

Abstract	7
List of Submitted Works	8
0. Introduction	10
0.1 Aims of this Study	10
0.2 Background and Rationale	13
0.3 Methodology	20
0.4 Literature Overview	22
0.5 Project Overview	29
1. Phase One	33
1.1 Introduction to Phase One	33
1.2 Full Bleed A/V	35
1.2.1 <i>Object Oriented Noise</i>	35
1.2.2 <i>Conceptual Spaces</i>	35
1.2.3 <i>Sound Objects/Performing Physics</i>	37
1.2.4 <i>The Show</i>	38
1.2.5 <i>The After-show</i>	41
1.3 Gravity	43
1.3.1 <i>If in Doubt</i>	43
1.3.2 <i>The Audience Perspective</i>	43
1.3.3 <i>HEAT DEATH INFINITY SPLITTER</i>	44
1.3.4 <i>Gravity</i>	49
1.3.5 <i>Reflections</i>	52
1.3.6 <i>Phrase Maker</i>	53
1.3.7 <i>Conclusions</i>	55
1.4 Live Coding	56
1.4.1 <i>The Lure of the Code</i>	56
1.4.3 <i>Going Live</i>	59
1.4.4 <i>Invisible Coding</i>	62
1.4.5 <i>Conclusion</i>	65
1.5 White Noise Festival	68

1.5.1 An Accidental Research Question	68
1.5.2 Pragmatic Strategies	69
1.5.3 MIDI Clip Wrangling	70
1.5.4 WNF.Output	71
1.5.5 WNF.Reflections	71
1.5.6 WNF.Conclusions	72
1.6 Music From Game Engines	73
1.6.1 Unity Triptych	73
1.6.2 EGX.Project	73
1.6.3 EGX.Output	74
1.6.4 MONOSATT.Project	76
1.6.5 MONOSATT.Output	77
1.6.6 MONOSATT.Conclusions	78
1.6.7 Post-Keyhole.Project	79
1.6.8 Post-Keyhole.Output	80
1.6.9 Post-Keyhole.Conclusions	81
1.7 Chapter Summary	82
2. No Man's Sky	87
2.1 Introduction to No Man's Sky	87
2.1.1 What is No Man's Sky?	87
2.1.2 Project Overview	88
2.2 All That is Solid...	93
2.2.1 Variform	93
2.2.2 The Main Album	95
2.2.3 The Album Soundscapes	96
2.2.4 The Pulse Soundscapes	99
2.2.5 The Pathfinder Soundscapes	105
2.3 No Man's Sky Live	106
2.3.1 What Happens in Vegas	106
2.3.2 Sonar	108
2.3.3 2Fly Studio Session	112
2.3.4 Tour	114
2.3.5 Reflections on NMS Live	115

2.4 Reflections on Infinity	117
<i>2.4.1 Chronotypology</i>	117
<i>2.4.2 Inevitably, it's Hegel</i>	118
2.5 Chapter Summary	122
3. Phase Two	124
3.1 Introduction to Phase Two	124
<i>3.1.1 Too Much Thinking</i>	124
<i>3.1.2 NaN Bpm</i>	125
<i>3.1.3 Transcending Atemporality</i>	126
3.2 Fully Automated Luxury Composition	128
<i>3.2.1 FALC in Context</i>	128
<i>3.2.2 FALC Reflections</i>	129
3.3 Zombie Coding	130
<i>3.3.1 Zombie Coding in Context</i>	130
<i>3.3.2 Further Reflections</i>	132
3.4 Studio Based Music Generation	134
<i>3.4.1 Intuitive Noise</i>	134
<i>3.4.2 Leaves</i>	134
<i>3.4.3 Scout & GoN</i>	136
<i>3.4.4 Backstage</i>	138
3.5 ZGEN - Generative Live Performance	139
<i>3.5.1 Live Collaboration With Algorithms</i>	139
<i>3.5.2 Z1</i>	141
3.6 Chapter Summary	143
4. Decomposition Theory	144
4.1 Composing Decomposition	144
<i>4.1.1 Ideas/Manifesto</i>	144
<i>4.1.2 Development Process</i>	145
<i>4.1.3 The Practice</i>	147
4.2 Technical Breakdown	149
<i>4.2.1 Euclidean Beats</i>	149
<i>4.2.2 Curved Beats</i>	154

4.2.3 <i>Mountain</i>	155
4.3 Performances, Stage Setup and Roles	158
4.4 Reflections	162
4.4.1 <i>Decomp in Context</i>	162
4.4.2 <i>The Question of Visibility</i>	163
4.5 Chapter Summary	165
5. Post Phase	168
5.1 Introduction to Post Phase	168
5.2 Decomp Futures	169
5.3 Nines	172
5.4 Budapest Live Show	174
5.5 65days.OS, KMF and A Year of Wreckage	177
5.6 Chapter Summary	182
6. Conclusion	184
6.1 Summary	184
6.2 A Critical Framework for Composition	187
6.3 A New Means of Disseminating Work	190
6.4 Theoretical Underpinnings of Decomposition Theory	192
Appendices	197
Appendix I: 65daysofstatic Discography	197
Appendix II: Full Bleed	197
Appendix III: YrCodelsBadAndYouShouldFeelBad	198
Appendix IV: Fully Automated Luxury Composition	198
Appendix V: Zombie Coding	198
Appendix VI: Behaviour Tree Notes	198
Appendix VII: Future Music Interview	199
Bibliography	200

Abstract

Bands as social and cultural forces primarily release albums and go on tour. These well established, popular forms for recording, presenting and performing music seem somewhat antiquated, similar now to how bands acted fifty years ago. This means that bands are not necessarily able to properly embrace new musical potentials that open up as technology evolves.

The goal of this study is to establish how new forms of expression and alternative modes of composition can be utilised by bands and solo artists working in the commercial popular music industry. It aims to do this through a practice-based exploration that specifically embraces unique or unconventional approaches to composition in the context of popular music.

The algorithmic and interactive strategies for music making that are focussed on in this study have been well covered in the academic literature, as have critical histories surrounding the evolution of popular music over the last century, up to and including its current moment. However, so far nobody has explored how emerging technologies might be applied to popular music forms from the perspective of an active participant working at a professional level in the music industry. This is what makes this study unique and significant.

To answer these questions, this study describes a new methodology. *Decomposition Theory* is a critically aware approach to music-making that sees composition as a dialectical process based around constructing musical and audio-visual systems that have the ability to generate endless musical potentialities. Established forms for popular music, such as songs, albums or live performances, can then be ‘decomposed’ from these systems. This is demonstrated through a collection of practice-based research at differing scales, from custom algorithmic functions generating individual phrases and rhythms at the micro scale, to large, audio-visual live performances, infinitely-long generative video game soundtracks, and new means of disseminating musical and audio-visual projects digitally at the macro scale. A theoretical grounding for the methodology is included in this accompanying written commentary.

This study finds that by bands putting *Decomposition Theory* into practice — that is by developing a new critical awareness to their own working practice, and viewing composition as a dialectical process rather than measuring it through completed musical productions — they need not be tied to established popular musical forms, but can discover new modes of reflection, composition, and alternative means for disseminating their work.

List of Submitted Works

Chapter One: Phase One

Audio:

- 1. Satt
- 2. Monos
- 3. MidiFlood
- 4. Heat Death Infinity Splitter (Excerpt)
- 5. Gravity
- 6. Gravity Phrase Examples
- 7. Live Coding Rehearsal
- 8. Flatland
- 9. Tidal Test (Excerpt #1)
- 10. Tidal Test (Excerpt #2)(Edit)
- 11. White Noise Festival Set
- 12. EGX Soundscape Capture
- 13. MONOSATT Capture
- 14. Post-Keyhole Capture

Video:

- Full Bleed A/V Live Performance
- Full Bleed Screen Capture
- MidiFlood PreVis
- Heat Death Infinity Splitter Live
- Gravity Demo
- Live Coding Rehearsal
- White Noise Festival Screen Capture
- EGX Talk
- MONOSATT
- Post-Keyhole

Chapter Two: No Man's Sky

Audio:

- No Man's Sky: Music for an Infinite Universe Disc 1 [album]
- No Man's Sky: Music for an Infinite Universe Disc 2 [album]
- In-game Soundscapes [audio collection]
- Sonar Rehearsal Recording

Video:

- The Games Awards Live Performance
- Supermoon Gameplay Trailer
- A Night Under No Man's Sky
- Sonar Live Performance
- 2fly Studio Session

- Asimov & Supermoon Live in Hong Kong

Chapter Three: Phase Two

Audio:

- 1. Fully Automated Luxury Composition
- 2. Zombie Coding
- 3. Leaves I
- 4. Leaves II
- 5. Scout
- 6. GoN
- 7. Backstage
- 8. ZGEN Live Performance
- 9. Z1

Video:

- Zombie Coding
- ZGEN Visuals and Drums Test

Chapter Four: Decomposition Theory

Audio:

- 1. Decomposition Theory Live
- 2. Rehearsal 250917
- 3. Euclid Off Kilter
- 4. Curved Beats: Skeletal Version
- 5. Curved Beats: Live Version
- 6. Mountain Render 260917

Video:

- Decomposition Theory Development Process
- Decomposition Theory Live Show

Chapter Five: Post Phase

Audio:

- 1. Looped Future
- 2. Strange Attractor
- 3. Nines
- 4. Budapest Live Show

Video:

- A38 Interview & Live Performance
- 65daysOS Archive
- KMF

0. Introduction

0.1 Aims of this Study

For this study, the practice is the research. The original contribution found within is the music and video that this written commentary exists to support. This practice is the result of an emic study undertaken by an active participant in the music industry, not by an outside observer.

The primary aim of this research is to build a portfolio of works that answers the question: how can new forms of expression and alternative modes of composition be utilised by bands and solo artists working in the commercial popular music industry?

As this is an emic study with a partly autoethnographic approach that is explained in the methodology, the solo artist in question is me and the band is 65daysofstatic, the group I co-founded and have been a member of since 2001. Both myself as a solo artist and 65daysofstatic are currently active in the music industry and are subject to the various creative preferences and economic pressures associated with that domain. These influences include a preference for intuitive rather than reflexive composition, extra-musical aesthetic and logistical concerns that bands and solo artists encounter, and the subtle and not-so-subtle economic forces that influence the output of a working band in popular music culture, all of which will be discussed throughout the study.

The new forms of expression and modes of composition that this study is concerned with are reflexive, documented and theoretically-underpinned creative processes that are not necessarily cutting edge in and of themselves, but are processes not conventionally found in the music industry in the context of a band like 65daysofstatic or a solo artist like myself (and if they are found at all, they are not accompanied by academically rigorous commentary from the participants themselves). They include

O. Introduction

working with technologies such as video game engines and algorithmic music frameworks as well as composing music for sound installations, interactive soundtracks, mobile phone apps and large scale, generative A/V performances.

There is much existing literature on the various disciplines covered in this research. For example, algorithmic music theory is a fast moving topic with a lively academic discussion that takes place across a range of journals including *Organised Sound* (Miranda, E., & Williams, D. 2015) and *Leonardo* (Magnusson, 2011), and is increasingly being discussed in journals that are concerned as much with the social relations of music as they are the mechanics behind it, such as *Dancecult* (Armitage, 2018) and *IASPM* (Wolinski, 2017). A summary of algorithmic music theory will be covered here in the literature overview and specific work will also be cited throughout this commentary, but it is important to note from the outset that although a significant amount of work here draws upon algorithmic music techniques, this is not intended as a study on algorithmic music. Similarly, the theoretical framework for this practice draws upon cultural theorists like Mark Fisher, Fredric Jameson, Adam Harper and Jacques Attali, who construct solid historical accounts and definitions of relevant music genres and associated theory, and write convincingly about what has happened, posit why it might have happened and speculate on what could come next. This thesis is not intended as a comprehensive overview of popular music studies, however these authors and their ideas are covered first in the literature overview and subsequently throughout the study to contextualise the relevant practice.

For this study, the practice itself is the research and its original contribution lies in the fact that this work is undertaken from a position of a true insider and active participant who is able to directly document how these new approaches to music can be utilised in the context of a band working in the contemporary music industry, and who can reflect on how a theoretical, academic perspective and research-based approach to composition,

O. Introduction

when applied to what was previously an intuitive and heuristic practice, can help (and sometimes hinder) musical production.

0.2 Background and Rationale

In 2001 I co-founded a band called 65daysofstatic. As of the time of writing in 2019, the band has released seven full length albums and performed live around the world many hundreds of times. A full discography can be found in Appendix I. We are an instrumental band made up of four members: myself (piano, guitar, programming), Rob Jones (drums), Joe Shrewsbury (guitar, keys) and Simon Wright (bass guitar, keys, programming). In terms of organisation, we make decisions collectively. We each have our preferred instruments and nominal musical roles in the band, but as time passes these distinctions are becoming increasingly deconstructed.

Our first record, *The Fall of Math* (2004), was released in 2004. Since then we have been making albums and going out on tour. Over time, our approaches to the studio and our live show have grown increasingly distinct. Although both generally feature the same material, over time the manner in which we recorded any given song diverged compared to how we subsequently performed it. When considering a song for inclusion on an album, the specific roles of who played what became less important, while sound design came to the fore. In the studio we became four co-producers rather than four musicians. In contrast, when building a version of a song for a live performance, a song's dynamics might be exaggerated, or the mixes changed. For example, extra sub frequencies might be added to the kick drums, or parts might be split or combined between each of us to make a performance satisfying both for us as musicians and also from the perspective of the audience.

As we established different approaches to these two forms through which we released or performed music, it made us want to explore what other forms there might be, forms less common for bands to be seen using. So far for 65daysofstatic this has included working and performing with a contemporary dance choreographer (2010) and

O. Introduction

playing live with a scientist while soundtracking a short documentary about the space shuttle missions (2012). We re-scored the 1972 science fiction film *Silent Running*, toured it around Europe then released it as an album (2011). Around the release of our 2013 album *Wild Light* (2013), we built a sound installation called *Sleepwalk City* (2013), based on one of *Wild Light*'s songs. *Sleepwalk City* ran autonomously as a 16-speaker audio-visual installation through the day, and during the evening it transformed into a performance space for a series of surround sound audio-visual sets by the band. This *Wild Light* campaign also saw us produce and release experimental short films (2014), collaborate on procedurally-generated music videos (2013) and strive to get essays published in the music press in addition to more regular interviews (Wolinski, 2014).

In terms of musical influence, there are some key examples in popular culture at the turn of the millennium that are illustrative of the kind of music and artists that shaped the early path of 65daysofstatic. Radiohead's *Kid A* (2000) was a record that saw the band leave behind their guitars and take a leap into the kind of cold, mechanical electronica more usually associated with acts like Autechre or labels like Warp Records. This departure from conventional band instrumentation and willingness to experiment was made more impressive given the position Radiohead held in mainstream popular culture. The Burroughs-style audio cut-ups and samples that opened the otherwise guitar-heavy album *Madonna by ...And You Will Know Us By the Trail of Dead* (1999) along with the curious interludes and bombastic production of Ross Robinson on At the Drive-In's *Relationship of Command* (2000) showed us bands who were able to combine their raw, post-punk, angular guitars with rich-sounding, often subtle, sonically interesting sound design on record. At the same time they presented themselves as visceral, hurricane-like forces of energy while on stage.

From an electronic music perspective, artists like Kid 606 and Cex from the Tigerbeat6 label were pioneering a new glitch aesthetic. These artists were well placed to

O. Introduction

take advantage of realtime digital audio processing and synthesis becoming available on consumer priced laptops for the first time. This technological development led to an explosion of music software and ways of integrating computer music with live bands in ways that had previously been either impossible, or only available to already successful artists with access to expensive professional audio equipment.

This was the context in which 65daysofstatic started performing live. For the very first performances, when we were still a three-piece without a drummer, the laptop we used onstage was not capable of handling realtime audio, and was instead used to run Cubase as a MIDI sequencer. This sequencer was driving samples from an Akai S2000 sampler and synth voices from a Korg M1. In addition to that, there were two guitars and one bass guitar. All the rhythms came from the sampler.

As the band evolved in those early years, so too did consumer technology and music software. Ableton Live was released in 2001, but as the 65daysofstatic live show made such heavy use of MIDI, it wasn't until MIDI clips were added to Ableton Live 4 in 2004 that we made the switch to using this. As Ableton Live developed, it became more tightly integrated into the band's live performance. As our back catalogue grew, more songs in our live set used audio loops and clips alongside MIDI. We started to add more MIDI controllers on stage, either keyboards or MPC-style pads, and started running more software synthesisers and samplers inside Ableton, reducing our reliance on external hardware.

In terms of composing, 65daysofstatic continued using Cubase until 2008 when we switched to Apple's Logic as our main Digital Audio Workstation. This lasted until 2013's *Wild Light*. Around this project, working on pieces like the *Sleepwalk City* installation led to using Max For Live for more specialised control over audio, and as we hit limitations within that we switched to using the fully-fledged version of Max, usually still running in conjunction with Ableton Live.

O. Introduction

In 2014 we started work on *No Man's Sky* (Hello Games, 2016), an ambitious project that required 65daysofstatic to write huge libraries of nonlinear music to make an 'infinitely-long', interactive soundtrack. *No Man's Sky* has its own chapter in this thesis. From the beginning of this project, Ableton Live became the primary piece of software for music composition within the band because of its modular nature, ease of integration with Max and the speed with which we could pull sketches together.

How this led to a research question

This short history of 65daysofstatic and our relationship with music technology provides the background rationale for this thesis. Since our inception, 65daysofstatic has evolved from being a band that makes records and plays live shows to a band that soundtracks, programs, and creates sound installations, audio-visual pieces, films, synths and software. Both myself and Simon Wright have released solo music that exists in its own right, yet feels like a subset of the band, part of the overall arc of 65daysofstatic's output. Our goal has always been and continues to be to expand the definition of what it can mean to be 'a band'. Furthermore, we have not only strived to find new forms to work within, but to explore the boundaries of already established ones. In particular, our live performances have been reinvented many times over the sixteen years we have been playing together. Whenever possible we have incorporated technological developments to expand our possibilities on stage. As this thesis will show, the question of how to perform live electronic music throws up many further questions, as well as problems, opportunities, and unexpected creative decisions.

The overriding research question of this study is: how can new forms of expression and alternative modes of composition be utilised by bands and solo artists working in the commercial popular music industry? In a sense this is a formalisation of what

O. Introduction

65daysofstatic had been doing anyway. The aim behind tackling this question with practice-based research is not only to explore new forms for 65daysofstatic to make or present music in, but also to look at how practice-based research itself in the context of a working band can offer new paths to creative expression or reflection.

This thesis is a practise-based research project, which is in part aligned with the trajectory of the band. Sometimes the two paths were one and the same, sometimes they ran parallel to each other, never actually touching in any easily quantifiable way beyond me being involved in both; occasionally they diverged. The music made during this study is the primary research content of this project and makes up the accompanying portfolio; this writing complements the music with commentary and contextualisation, and reflects on both my experiences and the research itself.

When talking about creative production in the thesis I will use ‘I’ when the work was primarily undertaken by me alone or if this is my own opinion, experience or reflection on a subject. I will use ‘we’ when discussing a 65daysofstatic project or decision made collectively. These terms imply a distinction that, in reality, is not as concrete as this language would have it. My account of 65daysofstatic is only one perspective out of four, all of which are equally valid. Furthermore, when I talk in the first person, my thoughts and ideas may well be shared by all four of us, and likely only emerged as products of our conversations and working together for so long in the first place. When I talk about my approaches to composing without referencing a direct work, this is as likely to apply to my role as a quarter of 65daysofstatic as it is my solo work. In many ways, this entire PhD research project is a subset of my role in 65daysofstatic, undertaken as much to specifically further that cause as to produce this body of work. However, it is me that has chosen to explore the subject academically, and feed back results into the band’s work. This PhD is a solo project, but one that is carried out in the context of membership of a popular music band.

O. Introduction

Before beginning this research, when thinking about the wider implications of existing and working as a band in the popular music industry, inspiration was taken from Mark Fisher in particular, whose work maps the connections between the worlds of contemporary music, politics and philosophy. As will be seen in the literature overview, this cartography, articulated primarily by Fisher's *Capitalist Realism* (2011), gave me the means to orientate myself within a landscape I had been travelling and making music in for more than a decade. It allowed me to contextualise my practice of the last fifteen years to better grasp the historical and philosophical foundations that I had been intuitively but haphazardly building upon in all my previous work. In a way, this knowledge had always been there, but was enmeshed in my musical output, inseparable from its host. It was intuitable enough for me to have been able to develop as a composer through my practice, but ultimately unknowable and un-shareable.

During the period before this research project, but when starting to consciously apply more theoretical ideas to our music, we made *Wild Light* (2013). This album was a response to an ambivalence toward our current situation inside of the music industry, and the music industry's role in society at large. Previously, 65daysofstatic were driven by a stubborn, deliberate naivety about the idea that there was a new kind of music yet to exist and that we might be able to chase it down if we moved fast enough. In contrast, *Wild Light* dared to stop searching for a new, unthought-of approach to music making, at least within the context of the album form, concentrating instead on writing music that resonated within it as effectively as possible.

In a later written review of Fisher's book *Ghosts of My Life* (2013) for the music website *The Quietus* (Wolinski, 2014), I made the connection between our reasons for taking this turn and what Fisher called "The Slow Cancellation of the Future" (2013). This was a first tentative step to what ultimately led to beginning this research project. A quick sense of my pre-research thinking:

O. Introduction

For years now there's been this nagging feeling that things have somehow become stuck, especially in terms of electronic music – although not just that. William Gibson, Bruce Sterling and that Sci Fi futurist crowd like to refer to it as 'atemporality'. Mark Fisher approaches it using Derrida's concept of hauntology and, in his introductory essay 'The Slow Cancellation of the Future', sets about sharpening the edges of these ideas that had been floating round my mind, fuzzy and unformed... I didn't think these ideas were unique to me by any means, but until reading this book I lacked the means to properly communicate them even to myself, never mind anyone else. (Wolinski, 2014)

The building of a critical framework around this “nagging feeling” is what makes up this thesis, a practise-based research imbued with a self-awareness, with a consciousness of itself, in a way the back catalogue it follows was not. In becoming conscious, the practice transcends the traditional musical containers it is placed in and reveals itself as an ongoing process of work and reflection folding back in on itself.

0.3 Methodology

As is now established, I am the active participant creating this research project; a musician specialising in electronic music production and song writing, working both as a solo artist and in the context of a band that is active in the popular music industry.

A fusion of approaches are used to conduct this research. Based on distinctions set out by Candy (2006), projects in the thesis such as *Decomposition Theory*, *Leaves* and the output from *No Man's Sky* are practice-based research, as the research content is epitomised within the outputs. Other projects, such as *Gravity* and *Live Coding* are practice-led, since they focus on processes undertaken, which bring valuable insights to answer the research questions, even though the resulting practice may be subjectively found to be lacking, as they are of lesser interest than what is learned. All the projects contain an element of autoethnography (Ellis et al, 2011). *Zombie Coding* is a fully autoethnographic written manifesto, *Decomposition Theory* and *No Man's Sky* contain excerpts of in-situ notes made during their respective development processes, and throughout this commentary every piece of practice ends with a subjective, critical analysis of the practice that is then used to formulate a new research question for the following piece of work to try and answer. Unlike some autoethnographic research projects, for example the website for Colin Webber's doctoral thesis *Creating a Virtual Heart* (Webber, 2009), the autoethnographic text here is not separated from the more formal writing. It instead follows each piece of practical work to maintain the conceptual path taken through this study as the practice evolves, with the autoethnographic response to each piece informing the research question the next piece tries to answer.

The approach used here is somewhat autoethnographic. Although the exact method for each piece of work differs in the details, the general form is that in order to answer a research question, a musical system or framework is built within which I or

0. Introduction

65daysofstatic can compose or perform music, or to which I (or we) can give input so the system composes or performs autonomously. My own subjective experience of the output of these systems is then considered from the perspective of the composer-participant, and this reflection is fed back into the musical system. This is an iterative process that remains subjective but is nevertheless, in the case of autonomous or generative musical systems that produce unexpected results, simultaneously bound up with the experience of an observer.

Instead of including a comprehensive literature review at the beginning of this thesis, the relevant concepts and theories are cited and discussed in detail throughout the commentary in respect to specific pieces of practical research. To give an outline of the breadth of philosophy interrogated in this study from the outset, there follows a short literature overview.

0.4 Literature Overview

The broad topics of literature that are relevant to this study are as follows:

- Algorithmic and Generative Music
- Capitalist Realism, Postmodernity and Inventing the Future
- Video Game Music and the Unity Game Engine

Algorithmic and Generative Music

Algorithmic music generation is by no means a new phenomenon. Mozart was experimenting with automated composition more than two hundred years ago (Alpern, 1995). As composers like Xenakis began experimenting with computer-based algorithms in the 1960s, automated composition gravitated toward the domains of avant-garde music and computer science, with research into areas like the processes of procedurally generating music and methods of simulating musical creativity (Dodge & Jerse, 1997).

Brian Eno, who coined the term *generative music* (Intermorphic, 2019), played a major role in bringing the practice of composing nonlinear, endless pieces of music into popular culture. His ideas appear throughout this commentary in relation to various approaches and creative decisions made in this research. Notable works by Eno that are relevant include *Discreet Music* (Eno, 1975) and his recent generative album *Reflection* (Eno, 2017) that was released in the form of a mobile phone app.

Over the last fifteen years, alongside the technological progress that allowed computers to start synthesising algorithmically-generated audio in real time,

O. Introduction

researchers in live coding have been making concerted efforts to build software and tools to create gentler learning curves that allow non-coding musicians to get involved in algorithmic music and therefore move this practice beyond academia and into the realm of electronic dance music culture. This discipline, whose early days are well documented by Nick Collins et al in *Live Coding in Laptop Performance* (2004), Thor Magnusson in *Herding Cats: Observing Live Coding in the Wild* (2014) and the comprehensive *The Oxford Handbook of Algorithmic Music* (Dean & McLean, 2018), is succeeding in broadening its horizons. Issue 1, volume 10 of the journal *Dancecult* from 2018 was dedicated to algorithmic music and a short documentary about the live coding ‘algorave’ community was produced by Resident Advisor magazine and featured prominently on the Resident Advisor website (Resident Advisor, 2019).

Capitalist Realism, Postmodernity and Inventing the Future

As discussed earlier, the work of Mark Fisher was an important starting point for this research. He describes his concept of *Capitalist Realism* as “a pervasive atmosphere, conditioning not only the production of culture but also the regulation of work and education, and acting as a kind of invisible barrier constraining thought and action” (Fisher, 2011). As a process, capitalism requires endless growth and is thus designed to subsume everything in its path, and re-present it with an intrinsic monetary value. Fisher argues that it becomes close to impossible to conceive of possibilities outside of capitalism, because everything that we have direct experience of exists inside that system and therefore this mode of existence

O. Introduction

closes down possibilities in all forms of human expression. Applying these ideas to popular music, Fisher describes Kurt Cobain as a tragic subject in this process:

In his dreadful lassitude and objectless rage, Cobain seemed to give wearied voice to the despondency of the generation that had come after history, whose every move was anticipated, tracked, bought and sold before it had even happened. Cobain knew that he was just another piece of the spectacle, that nothing runs better on MTV than a protest against MTV; knew that his every move was a cliché...Cobain found himself in 'a world in which stylistic innovation is no longer possible, [where] all that is left is to imitate dead styles, to speak through the masks and with the voices of the styles in the imaginary museum'. (Fisher, 2011)

From Fisher, reading for this thesis branched out in various directions. Most immediately, Fisher's capitalist realism is derived in part from the writing of Fredric Jameson, who equates this cultural logic of late capitalism to postmodernism. A large body of literature covers all aspects of postmodernism, of which Jameson's work is just one perspective of many. What is relevant to this research's focus on finding new forms for creative expression are the efforts to move beyond the "end of temporality" (Jameson, 2003) that postmodernity has brought about. Jameson's notion of a "postmodern artistic singularity effect" (Jameson, 2003), one possible route to making art that escapes or at least embraces postmodernism's reductive, referent nature, is explored later in this writing. In *Inventing the Future* (2015), Srnicek and Williams suggest that a vital role that falls to artists in the contemporary moment is to imagine better futures, because only once we can imagine what a better future might look like are we able to start building it. Montuori sees the future of creativity after modernism as becoming more "contextual, collaborative [and] complex" (Montuori, 2010), embracing generative techniques and new technologies. Adkins' *A Manifesto for Nodalism* (2014) draws

upon, amongst other things, Bourriard's desire for an "assumed heterochrony" (Bourriard, 2009) from which we can start to realise an art that is no longer already-drowned in signs and referents. This is from Bourriard's concept of altermodernism, which is also featured in *Supplanting the Postmodern* (Rudrum & Stavris (Eds.) 2015), a collection of approaches concerned with new ways of experiencing reality that includes a chapter on performatism, a concept that is explored further in the *Zombie Coding* practice found in Chapter Three. Finally, Jameson's writing in *Valences of the Dialectic* (2009) and *Archeologies of the Future* (2005) are relevant here in the context of examining how the utopian impulse can be used by composers to take tentative steps toward imagining new forms for creative expression. This is looked at in depth in *Fully Automated Luxury Composition*, also found in Chapter Three.

Another branch of literature that stemmed from *Capitalist Realism* (Fisher, 2011) is a thread of Marxist thought that can be traced back from Fisher through Jameson's work, Jacques Attali's *Noise* (1985) that examines the political economy of music and is of particular relevance here in regard to its speculative look at future modes of composition, Theodore Adorno's views on "the Culture Industry" (Adorno, 2005) and popular music's role in it, all the way back to Marx himself. *Music and Marx* (Qureshi, 2013), with a foreword from Attali, is a collection of essays that take various approaches to engaging with the ways in which Marx's views on the commodity form and social relations intersect with music composition. Relevant here is *Commodity-Form, Disavowal and Practices of Music Theory* (Klumpenhower, 2013) which looks at how to avoid "the fetish[ised] nature of institutional music theory" (Klumpenhower, 2013) and analyse music as a

product of social relations rather than some ephemeral piece of art that exists separate from the world, and *Modernity and Musical Structure: Neo-Marxist Perspectives on Song Form and Its Successors* (Manuel, 2013), which applies a Marxist critique to the very idea of a song itself.

As Marx said, “It is not the consciousness of men that determines their being, but, on the contrary, their social being that determines their consciousness” (1859). Though not explicitly Marxist, Adam Harper’s *Infinite Music* (2011), Robert Barry’s *The Music of the Future* (2017), Simon Reynolds’ *Retromania* (2011) and Christopher Small’s ideas about using “musicking” (2011) as a verb to encompass all the ways and relations in which people experience music, collectively analogise Marx’s idea here to suggest that it is not a consciousness of popular music that determines its being, but, on the contrary, the social being of popular music that determines its consciousness. As this body of literature that does not treat popular music as somehow inferior to what Adorno would call “serious music” (1941) grows, it builds a framework for studying popular music that accepts that the music does not exist in a vacuum of classical music theory, but rather it exists within and through society.

Video Game Music and the Unity Game Engine

Despite the speed at which the video games industry is evolving, standard practises in video game music are nevertheless beginning to emerge both in academic discussion and game audio middleware like FMOD and Wwise. As these approaches become observable, so too do the limits of applying classic

critiques and compositional methods to an undeniably new form. For the composer, the dialectic between the writing of a fixed score and the demands of a dynamic score that reacts to its environment offers two poles at either side and a spectrum of compromise in the middle. Any loosening of control over the arrangement, (whether temporally or spectrally) dilutes the preciseness of the piece of music being written.

Karen Collins is a leading voice in the field of video game music. *Game Sound: An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design* (Collins, 2008) and Winifred Phillips' *A Composer's Guide to Game Music* (Phillips, 2014) both provide comprehensive overviews of the various techniques used by composers to create soundtracks that can respond to the actions of the player. Meanwhile, conferences like Ludomusicology (Leeds, 2019) are bringing researchers from academia into contact with sound designers and musicians working within the industry. Further literature surrounding video game music that is relevant to this study is covered in more detail in the chapter on the *No Man's Sky* video game and the infinitely-long score that 65daysofstatic composed for it.

Unity is a game engine that is popular with smaller, indie developers. It allows developers to create games within a 3D space and includes many important frameworks generally required to make a game, meaning the developer does not have to begin from scratch. While Unity has an audio system that is in active development and that has improved significantly during the period of this research project, perhaps because of the speed of development, making music specifically within the Unity game engine is an area where there currently appears to be a gap

in the literature. While video game music compositional techniques are well covered by Collins and Phillips, the technical aspects of actual composing and recording they discuss tend to be done in conventional musical software like Logic, ProTools or Ableton Live, and then audio middleware such as Wwise or FMOD is used to integrate this music into the game. A reason why literature on Unity-specific music is lacking is that a lot of the techniques that can be used to compose within Unity itself are generic concepts and software design patterns and these would be covered more generally in computer music journals. In *Worlds of Sound: Indie Games, Proceduralism, and the Aesthetics of Emergence* (D'Errico, 2015), Errico draws parallels between generative music techniques emerging in video games and generative art techniques from the twentieth century that preceded them. The examples he uses, spanning Unity, Pure Data and going back to pre-digital ideas from Steve Reich and John Cage show that when it comes to algorithmic music concepts, they are very often platform agnostic.

Nevertheless, using game engines themselves to write music is an area deserving of further study, not least because they exist at the cutting edge of technology, allow responsive music to be deeply integrated into visual environments and open up new ways for potentially distributing music. Outside of academia, developers like Matt Tytel (2019), developer-artists like Keiijiro Takahashi (2019) and hardware companies like Teenage Engineering (2019) are all examples of an active community that create both commercial and open source frameworks that allow for coordinated music and video projects to be built entirely within the Unity engine.

0.5 Project Overview

This research was carried out in five phases which are separated into five chapters. The order of the chapters helps trace the development of technical and philosophical approaches to composition, and gives a loose impression of the general chronology of these phases. In reality there was overlap. The *No Man's Sky* (Hello Games, 2016) project in particular ran in parallel to the research described in the chapters called Phase One and Phase Two. This graph best describes when the projects occurred:



The Phase One chapter documents an initial run of musical experiments based on a deliberately broad approach to exploring algorithmic techniques in composition and looking at how integrating visuals could offer new perspectives to a composer. These first

O. Introduction

experiments primarily concentrate on harnessing software, protocols and programming languages that are either new additions to my compositional tools (Unity, the C# or ‘C Sharp’ programming language) or that had previously only existed at the peripheries of my work (Max, Open Sound Control, javascript). Woven into the discussion of these projects are the emerging theoretical underpinnings that develop alongside the practical work and help clarify and focus the research throughout this first phase.

The next chapter looks at 65daysofstatic’s involvement in writing the score to the video game *No Man’s Sky*. As can be seen, *No Man’s Sky* was an active project for a significant period of time (although as the chapter will show, there were large stretches of inactivity). Furthermore, *No Man’s Sky* was a project that had its own agency, it was undertaken in parallel to this research, was informed by this research, and this research in turn was informed by our work on the *No Man’s Sky* score, but at no point was that project a subset of this thesis. It nevertheless demands its own chapter here because, serendipitously, the soundtrack written by 65daysofstatic is a useful case study of a text that exists in multiple musical forms. It was released as a soundtrack album, it exists in the game as an infinitely long, ever-evolving collection of generative soundscapes that respond to the actions of the player, in the lead up to the release 65daysofstatic performed various live A/V shows that offer a glimpse into the pragmatic realities and commercial pressures that can shape creative choices, and after the release we toured around the world, playing music from the game in the context of their more usual live sets alongside our back catalogue.

Phase Two sees a purposeful tightening of focus in the practice, with the work scaling up to fewer but more involved projects that explore different facets of generative music techniques and their intersections with popular music concerns. *Fully Automated Luxury Composition* is research born out of approaching algorithmic music through the lens of utopian thinking. *Zombie Coding* is a confused experiment that begins as a

O. Introduction

deliberately performatist artist statement regards my live coding practice. Rather than succeeding in writing a statement about my practice, the piece of writing itself becomes a useful autoethnographical account that can be examined to see how my approach to composition evolves over the course of this PhD. Also in Phase Two, parallel to the above formalised research questions a concerted effort is made to re-apply the same techniques in a studio-setting, with less strict rules for documentation or critical analysis of the processes. For this material, the outcome of the practice is valued over the process. Lastly, *ZGEN* takes a look at an approach to live algorithmic performance.

Decomposition Theory is where all the strands come together. It is a manifesto, methodology and a live, algorithmic A/V set by 65daysofstatic. This chapter documents the project from early development to the debut live performance and speculates about possible futures based on these ideas.

The Post Phase chapter draws the research to a close. 65daysofstatic's work is ongoing and at some point a line needed to be drawn to limit and focus the scope of this research project. This chapter gives a short overview of work carried out after the end of this period as well as looking at the final practical work for this study, carried out in the aftermath of *Decomposition Theory*. Outside of the context of this study and commentary, this final practical work would most accurately be classed as practice-as-research, written as it was after a return to a more intuitive, heuristic approach to composition without clearly defined research questions in mind. Although none of this Post Phase work was approached specifically in the spirit of "seeking to add to the shared store of knowledge" (Candy, 2006), in the wider context of the thesis it remains useful to look at as all of it contains techniques or approaches that were explored in earlier research, such as algorithmic logic and tightly integrated visuals influencing the music. The Post Phase chapter contextualises and adds commentary to this work, serving as a useful bookend

0. Introduction

to this enquiry that shows how the research has changed my own and 65daysofstatic's approach to composition.

The final chapter concludes this project by summarising the analyses of all the creative production contained herein before positing a working strategy for musical composition based on a dialectical relationship with the various new techniques explored in this study. It also posits a dialectical relationship for the composer between reflexive, research-based approaches and intuitive, uncritical approaches to popular musical composition.

1. Phase One

1.1 Introduction to Phase One

The first stage of practical research looks at visuals being tied directly to the music, either as literal expressions of the music through projecting musical live coding, procedurally-generated interpretations of musical content driven by the same MIDI data as the music, or with the visuals themselves generating the music. By applying algorithmic techniques to my already existing approaches to composition at the beginning of this study, the intention is to reveal processes which can serve as guidance for more musically-focused algorithmic functions going forward. Additionally, this approach helps me become more familiar with certain pieces of audio-visual software, the protocols that allow them to communicate with each other and to start formalising my ideas about musical composition.

Some of the working practices that begin here can be followed throughout the three years and are still ongoing at the time of submitting this thesis. *Full Bleed A/V* serves as the means to conceive a software framework that allows visual software, custom code, game engine software and commercial music software to talk to each other. *Gravity* is the first piece of practical research made for an academic context and therefore embraced an experimental form that would have been unsuitable for public release in the context of my pre-academic musical practice. The *Live Coding* experiments test this burgeoning musical discipline for potential spaces it might open up for new expressions in the context of popular music. *White Noise Festival* is the result of all the experimental knowledge that had been so far gathered getting dragged mercilessly back out of academia and into the context of a working musician needing to build a new headline A/V show at short notice. The *MONOSATT* and *EGX* projects included in the

1. Phase One

Music From Game Engines section are slightly different as the visuals involved are not being primarily considered as part of a performance, but they both question how visuals can be utilised to re-contextualise or navigate the music and show the potential in applying game engines to musical composition. Finally, the *Post-Keyhole* project is a practical although ultimately unused application of the various techniques developed during this phase, with the goal of creating a generative music instrument that could be played (for reasons that become clear) using only one hand.

1.2 Full Bleed A/V

1.2.1 Object Oriented Noise

Full Bleed (Wolinski, 2014) is a solo recording made in the spring and summer of 2014 and released on limited edition cassette tape by Sacred Tapes in November of the same year. The audio can be found in Appendix II. That writing process is undocumented, but its undertaking was a conscious effort not to rely on the piano-roll-style, left-to-right linear composition that my previous work tended to be built on, usually put together using Logic or Ableton Live. It also had a heavier reliance on sound design than any previous solo work and, significantly, it was not written with any intention to be performed live.

In June, 2015, I was invited to play at *Sonic Pattern* (Wolinski, 2015), a small festival in a dark, loud arts space. I had no desire to perform old material, but also did not want to compromise the carefully produced *Full Bleed* sound design in order to make it performable live. This combination of happenings resulted in the following question: in the context of a performance-centric live setting, how can music be presented in a compelling way without including much in the way of live performance? *Full Bleed A/V* was thus developed in response to this invitation, integrating visual material in order to enhance the performance of *Full Bleed*.

1.2.2 Conceptual Spaces

Thinking about how to approach this performance led towards ideas of conceptual space and musical space. Practically, I began to use Unity as a compositional tool rather than as software that was nominally designed for building computer games.

Gärdenfors' theory of conceptual spaces allows for geometrical representations of knowledge and in *Unifying Conceptual Spaces: Concept Formation in Musical Creative*

1. Phase One

Systems (2010), Forth et al use musical examples to show how conceptual space can be useful in building artificial creative musical systems. Here they describe representing a melody in conceptual space:

Notes, though, are not enough to represent the experience of melody ... given such a representation [the relational pitch interval of notes in space], we can abstract further, to notions such as *melodic contour*, where we replace the pitch intervals with magnitudeless indicators of direction. (Forth *et al*, 2010).

When composing alone and with 65daysofstatic, thinking about music spectrally in terms of mixing, sound design and production is as important as the linearity, arrangement and ordering of the musical content through time. Conceptual spaces, made up of a number of qualitative domains that are "integral (as opposed to separable) dimensions" (Forth et al, 2010) each of which can have its own "geometrical, topological or ordinal structure" (Forth et al, 2010) produce a compelling image where musical events traversing through a sound space, such as a note moving from one pitch interval to the next, might pass through timbral or time domains that might otherwise be invisible and be thus transformed in unexpected ways. It suggests a de-quantisation or decomposition of conventional formal representations of music.

Adam Harper's idea of "music space" (2011) comes from the world of cultural studies rather than the world of computer science but provokes similar images. His point is that what composition can be is unbound. There are rules, conventions and genres that enable composition, but these are all just musical objects, subsets of an infinitely large music space.

We can start with a 'piano space' contained within music space: piano space is a subset of music space, and in turn a scale of A major played on a piano is a subset of piano space, the scale of A major starting at A4 played on a piano is a subset of that musical object, an A4 played alone on a piano is a subset of that musical object, and A4 played on a piano for four seconds is a subset of that (any of these musical objects could be members of other sets, of course, such as the set of all A major scales on any instruments) (Harper, 2011)

1. Phase One

Music space can contain countless musical objects that can stand alone, combine or overlap. Another term Harper uses for this space is “n-dimensional” (2011). That is, this space can be divided up however it needs to be and for composers (and to Harper this is not an exclusive term, we can all be composers) there still remains infinite music space left to explore. Harper’s full term is “n-dimensional modernism” (2011) and his idea is that to be modern is to search for previously undiscovered space:

Modernism is a directional process, the music it creates is always somewhere *between* the old and familiar and the indiscriminate infinity of different forms, proceeding only *toward* the latter. It’s a relation between old and new, and any given moment of modernist music will present a mixture of what can be appreciable to a given audience to any extent as either old or new. (Harper, 2011)

This direct idea of modernism, uncomplicated by any associations to the equally n-dimensional postmodernism that later stages of the research move toward, describes a clear, fundamental driving force of my creative practice: finding new ways of making music.

1.2.3 Sound Objects/Performing Physics

Inspired by the ideas put forward by conceptual and music spaces, a methodology to answer the question of how to present *Full Bleed* as a live performance became clear. It would be an A/V performance with visuals closely tied to the music, generating movement in 3D spaces that evoke the mood of each song. The audio would be mostly multi-tracked stems and the live performance element would be restricted to manipulating both analog effects (in the form of audio being sent through guitar pedals) and internal effects (in the form of a MIDI controller changing values of software effects in Ableton Live). The visuals were built as separate scenes within Unity and were controlled

1. Phase One

by Open Sound Control (OSC) messages being sent from a Max For Live (M4L) patch running in the main Ableton Live project.

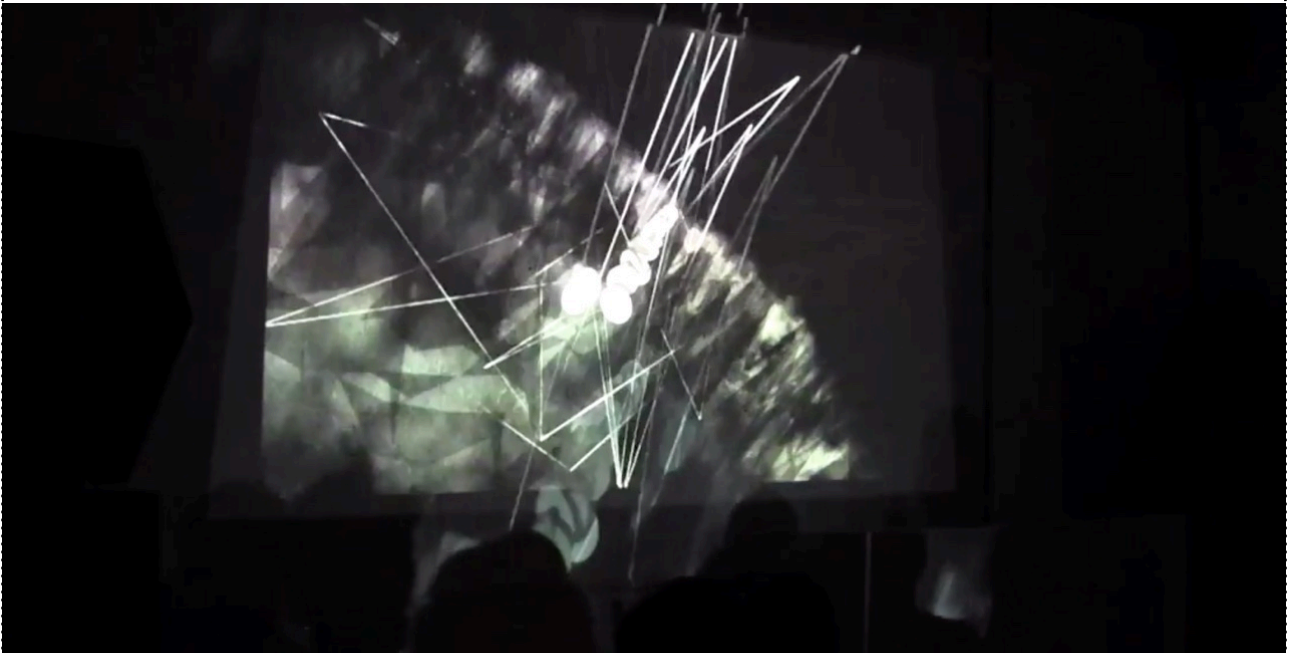
As the project came together, the OSC communication between M4L and Unity became bi-directional. This was to cautiously experiment with the idea of the visuals driving the music as well as the music driving the visuals. This can be seen in the live video from 2:18 - 4:00. The visuals show an empty plane above which spheres appear. The spheres are instantiated live by pressing pads on a MIDI controller. As the spheres succumb to gravity and collide with either the plane below or each other they send OSC messages back to Ableton Live, producing sounds. Although I am in control of instantiating spheres and also able to toggle the gravity in the 3D space on and off I do not otherwise have any control over the trajectory of the spheres. The physics engine of Unity is what creates the conditions for their collisions and the subsequent music that is made.

1.2.4 The Show

The work made for this project consists of three videos and three audio tracks. In addition to the original songs from the album, two additional songs were written alongside the building of the visuals. These are titled 'Satt' and 'Monos'. Also included here is audio of a song titled 'MidiFlood', which was not included on the *Full Bleed* album but written in the same period.

1. Phase One

Full Bleed Live Performance

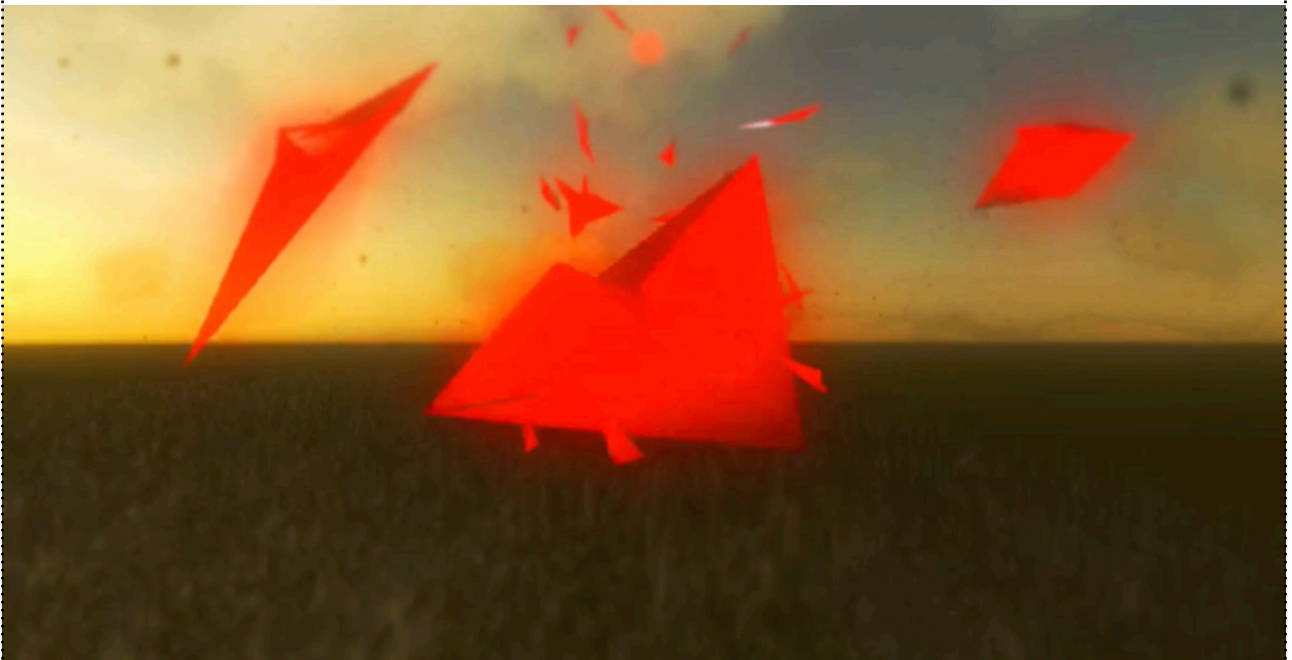


Camera footage of the debut show. The sound is a room recording.

Video Title: VIDEO_1_FullBleedAVLiveFootage

YouTube link: <https://youtu.be/9WTyziRUPI8>

Full Bleed Screen Capture



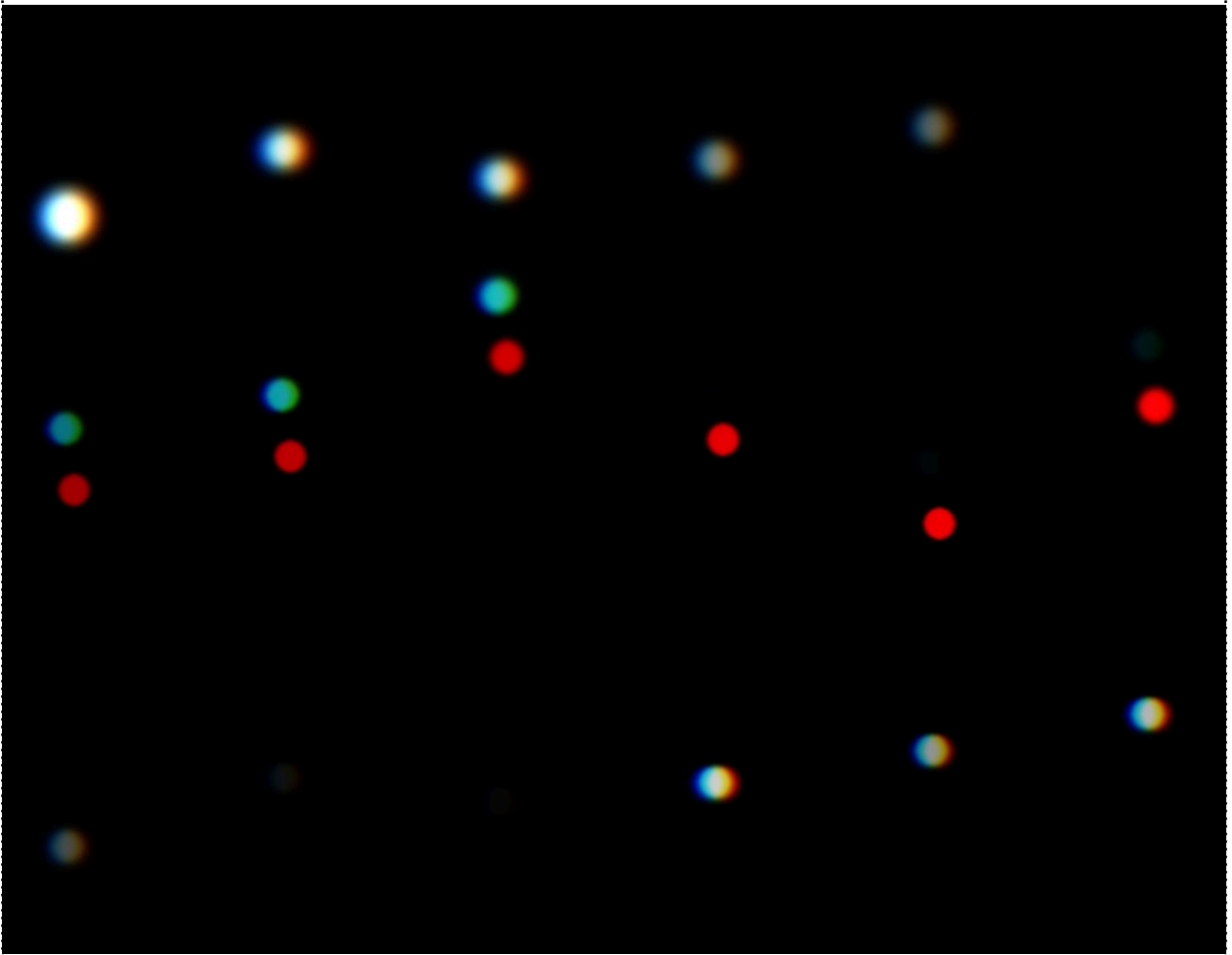
Screen capture of one of the final song of the set, 'Full Bleed'.

Video Title: VIDEO_2_FullBleedScreenCapture

YouTube link: https://youtu.be/SNB_XVXcoV4

1. Phase One

MidiFlood PreVis



Early test using MIDI to also trigger audio & visuals simultaneously.

Video Title: VIDEO_3_MidiFloodPreVis

YouTube link: <https://youtu.be/WJ9D2zB3DtY>

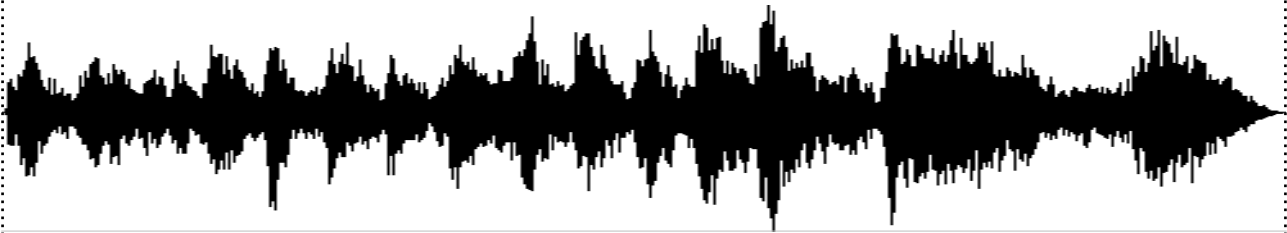
Satt (Chapter One, Track 1)



This track was written during the period of building reliable communication between M4L and Unity. Its unique feature is its fluctuating tempo track, which is controlled by a generative M4L device.

1. Phase One

Monos (Chapter One, Track 2)



This is noteworthy as it's a track that includes elements of live coding (see later in this section), but also because it was written in tandem with the visuals. This song was only played in later shows, so is not seen in the live performance video.

MidiFlood (Chapter One, Track 3)



This track was inspired by Steve Reich's "Piano Phase" (1967). A single piano line is played twice at two very slightly different speeds, via a Max patch. Given the construction of the song, with just four easily discernible strands of MIDI, it was a good test case to build visuals tied directly to the notes. The MidiFlood PreVis video can be compared to the Full Bleed Live Performance video from 11:25 to see how the visuals developed.

1.2.5 The After-show

Anecdotally, the show was a success. The crowd appeared to enjoy it and the sound was clear and loud. The biggest achievements for me were the various technical breakthroughs made to get multiple pieces of software working in concert. The bi-directional communication between Unity and Ableton Live/M4L exceeded expectations and became an invaluable design pattern as research progressed.

Although in the event everything worked as it should, by the time of the show its limitations had become clear. A predictable drawback, given the goal of making a minimally-performed performance, was that being onstage doing very little was an

1. Phase One

uncomfortable experience. Although the visuals were very much the focal point, after subsequent live shows in Liverpool (Wolinski, 2015), Manchester (Wolinski, 2015) and Derby (Wolinski, 2015), this lack of tangible interaction with the music was the overriding reason for not pursuing further performances.

A significant limitation was that the show had been built inflexibly. Although the visuals were being generated live by the music, most of the visual scenes were designed to work in very specific ways and were by no means fully formed interactive, responsive sound spaces. They were rather virtual stage sets. Furthermore, the music had a carefully arranged, linear timeline, peppered with carefully positioned triggers to load up new 3D scenes to match music that had intentionally been written outside the usual quantised grids associated with Ableton Live. This meant that inserting, removing or changing material in the set was a complicated procedure. Another drawback was that the reliance on primitive 3D shapes moving around in 3D space produced a computer game-esque aesthetic that I felt was at odds with the mood of the *Full Bleed* audio and original graphic design of the album.

Full Bleed A/V fulfilled its limited scope. It produced a neat A/V show that I was satisfied with given the scale and context it was being performed in. It also showed that music can be visualised in 3D space in a variety of ways using a game engine and that the game engine in turn can affect the audio. This opened the way to new areas of study. For example, how might utilising the game engine software to apply turbulence, physics or other disruptions to this sound space lead to new expressions for music composition?

1.3 Gravity

1.3.1 If in Doubt

Gravity was primarily made for a lecture-recital titled *If In Doubt, Make it Loud* (Wolinski, 2015) at the (Per)Forming Art Symposium. It is an example of a generative live performance. The lecture argued that if research questions in the vein of ‘what approaches are available to the laptop soloist looking to perform live?’ are followed up with a question like ‘and does the audience care?’ it can help contextualise the practise. It can also serve as an important reminder that if no understanding of the mode of performance is communicated to the audience then it can risk being redundant.

Gravity was built on questions raised in the *Full Bleed A/V* piece about how manipulating sound space could be used as a compositional tool. Before getting into the work itself, looking at how 65daysofstatic take the audience’s perspective into account when performing, and giving an example of how we approach preparing electronic music for a live setting, will help to give some contextualisation. 65daysofstatic’s methods in these respects directly inspired the building of *Gravity*, and were also the focus of the lecture that preceded its performance.

1.3.2 The Audience Perspective

David Byrne's *How Music Works* (2012) paints a sprawling picture of all the facets he feels make up the idea of a band, far beyond simply releasing and performing music. It takes us through a winding history of his band, Talking Heads, describing it as an art collective enmeshed in its local arts scene, popular culture and musical technology, pushing against social norms to expand the definition of what being a band can be (Byrne, 2012).

Similarly, 65daysofstatic have always felt that being in a band is about more than just the

1. Phase One

music we release. It is also about our attitude, our public voice, the manifestos, the artwork, the touring, the music we finish but decide not to put out, the live show and so on. Like Byrne, there has also always been a desire to antagonise the form, to strive to become a band unlike any band that has gone before.

In terms of designing live shows, especially from the perspective of incorporating the electronic elements into our sets, the preference for us was always to prioritise whatever felt most effective from the point of view of the audience. A live show cannot only be about how an audience perceives it, this concern is but one example of an endlessly complex layering of social relations. Based on his ideas of musicking being a verb that encompasses all the ways in which people experience music, Christopher Small describes,

The way people relate to one another as they music is linked not only with the sound relationships that are created by the performers, not only with the participants' relation to one another, but also with the participants' relationships to the world outside the performance space, in a complex spiral of relationships, and it is those relationships, and the relationships between relationships, that are the meaning of the performance. (Small, 2011)

Although these relationships transcend the performance space, they are felt very directly in a live setting through the direct performer-audience interactions, and therefore thinking about the audience's perspective is one of the major distinctions when considering live representation of music as opposed to, for example, producing an album.

1.3.3 HEAT DEATH INFINITY SPLITTER

This live performance of “Heat Death Infinity Splitter” (65daysofstatic, 2013) provides an example of some of the specific considerations undertaken as a band when performing live.

HEAT DEATH INFINITY SPLITTER Live

1. Phase One



Footage of a live show in St. Petersburg shot from the crowd.

Video Title: VIDEO_4_HeatDeathLive

YouTube link: <https://youtu.be/0eanqfN52uM>

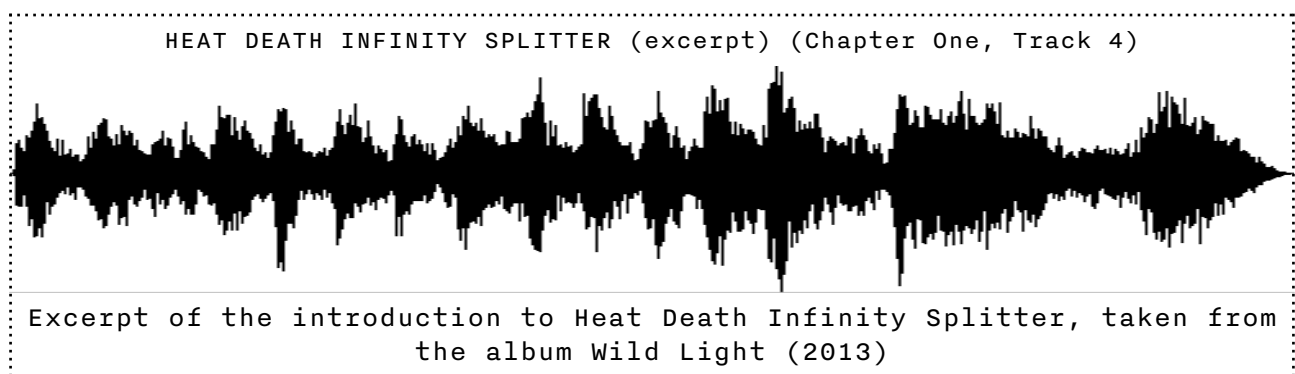
On stage right (to the left of the screen), I am playing a MIDI controller keyboard that drives four synthesisers mixed together. I have a click track in my headphones and play alongside a pre-made audio stem of a sine wave bass line. Joe (stage centre) is playing guitar. Simon (stage left) is performing a lead melodic phrase by triggering and pitch bending samples and Rob (stage rear) is triggering drum samples on v-drums. Although this might at first seem like a logical or arbitrary division of sounds, if we break it down further we can see it is a result of a combination of experience, technical decisions, and an understanding that the representation of a musical performance is not necessarily the same as simply performing the music.

Why was this particular combination of live instruments and playback chosen? There are other ways we could have chosen to perform this song. Simon, nominally speaking our bass player, could have played the bass line on a MIDI controller whilst the

1. Phase One

melody was playback. Or the guitar could have been sampled whilst Joe played the melody line. The sine wave bass synth could have been triggered by my left hand and transposed down an octave, which would have produced a part almost identical to the playback track.

In order to discuss the decision of the bass line not being played by Simon, consider this short excerpt of the recorded version of “Heat Death Infinity Splitter” (65daysofstatic, 2013):



The choices we made to produce this song in the studio were all geared toward a singular sound. If we try to describe it to each other, it is generally referred to as a wall of noise or something similar. What we actually mean about the sound we want to make at the perceptual level can be formalised using Wertheimer's Gestalt theory. A gestalt is something "not merely *more* than the sum of its parts, but fundamentally *different* from the sum of its parts...It is *prior* to its parts" (Wertheimer, 2014). This sound is not designed to be perceived as a combination of synthesisers, noise and guitars all playing a series of notes. It is designed to be the entire identity of “Heat Death Infinity Splitter”. A gestalt, an indivisible whole.

Playing this song live, this particular sound becomes the responsibility of myself on the MIDI piano and Joe (centre stage) on the guitar. As much as we chase the gestalt on record, the live performance brings with it other priorities. Rather than chase the sound achieved through careful sound design and layered studio recordings, we instead look for

1. Phase One

ways to visually represent the scale of it, by all four of us contributing to it in a manner that can be perceived by the audience. On the record, although guitars are used to create part of the noise in question, they are not necessarily recognisable as guitars as such. To recreate this sound live, technically, sampling these recordings and layering them with the synths would produce a more accurate representation. This could all be played on the MIDI keyboard, but what would then be lost is the physicality of the presence of a guitar. Denis Smalley describes any gesture involved in performing music as an “energy-motion trajectory which excites the sounding body, creating spectromorphological life” (Smalley, 1997). A performance is a collection of gestures, some of which are more extravagant than others. The relentless striking of guitar strings by a central figure on stage is a stronger, larger gesture than the finger movement over on stage right. Having the core sound provided by the MIDI keyboard, then adding live guitar, with all the distortion, feedback, physicality and excitement associated with it adds to what the audience is experiencing. The gestures more clearly appear to manifest the music. Smalley suggests that gestural music “is governed by a sense of forward motion, of linearity, of narrativity” (Smalley, 1997).

By the same measure, Simon could perform the four sine wave root notes of the bass line on a MIDI keyboard, or they could be layered up with my synthesiser so they were being triggered by my left hand, but neither approach would result in any discernible difference from the line being played back as pre-rendered audio as far as the audience could see. By letting this repetitive bass line be playback, it allows Simon to be free to instead perform the melody line, a phrase whose frequency is distinguished enough from the wall of noise to be heard as an individual element. Again, this doesn't help us manifest a gestalt sound, but instead increases the scale of performance as seen by the audience in connection with the sound. Having a sample-based, electronic melody clearly represented by a person at a keyboard helps the audience perceive what Smalley would

1. Phase One

otherwise call a remote surrogacy sound — a sound where “source and cause become unknown and unknowable” (Smalley, 1997) — instead as a second order surrogacy sound, associated with the keyed instrument they can witness Simon playing.

Furthermore, in terms of technical concerns, there are quantifiable positives by having the bass line run as pre-made audio. It means that there doesn't need to be another software synthesiser taking up more RAM on the computer set-up. It also means that this audio can be eq-ed and compressed with an awareness of its place in the overall frequency spectrum and routed to its own dedicated output to the front of house engineer.

The logical end point of that last argument for using playback is that everything is pre-recorded, eq-ed and mixed into a number of audio stems, so that the front of house engineer could mix them based on the acoustics of the room. In a purely acoustic sense, this is the path to the best sound it would be possible to produce and as a bonus we could ensure there were no mistakes. There is no written rule to say this is wrong. Concerts from the world of electroacoustic or acousmatic music often contain only pre-recorded material. Questions of authenticity, virtuosity and playback of pre-made audio in live electronic music are well covered in *Deadmau5, Derek Bailey, and the Laptop Instrument – Improvisation, Composition, and Liveness in Live Coding* (Parkinson and Bell, 2015). Deadmau5 once famously declared, seemingly on behalf of all electronic music performers that “we all hit play” (Zimmerman, 2013) and do not do much else onstage, whilst the authors themselves conclude that there is a spectrum of possibility for live electronic music with Deadmau5's approach at one end and live coding practices opening up new potentialities at the other.

For 65daysofstatic, unlike Deadmau5 or a typical live coding set, in addition to the electronic elements, in the example under consideration there are four members on stage with a range of physical instruments. We would not be willing or able to mime to an

1. Phase One

entirely pre-recorded performance, although this is common practice within some genres of commercial chart pop music. It would not be appropriate to our aesthetic and is far less common in, for example, indie and alternative forms.

The decision about how to perform the bass line to this song is just one of the numerous choices we made when putting together a new set. For us these choices cannot be generalised, each one has to be appraised through a combination of the specifics of the song in question, who played what on the recorded version (if a recorded version exists), whether the audience would actually notice a specific performance choice, and a host of pragmatic technical constraints such as the number of instruments we can bring on tour, or the amount of spare MIDI ports we have available in our live set-up, amongst other variables. Far from being simple, direct recreations of recorded music, our live performances are in fact a collection of many creative choices about how that music gets represented in front of an audience.

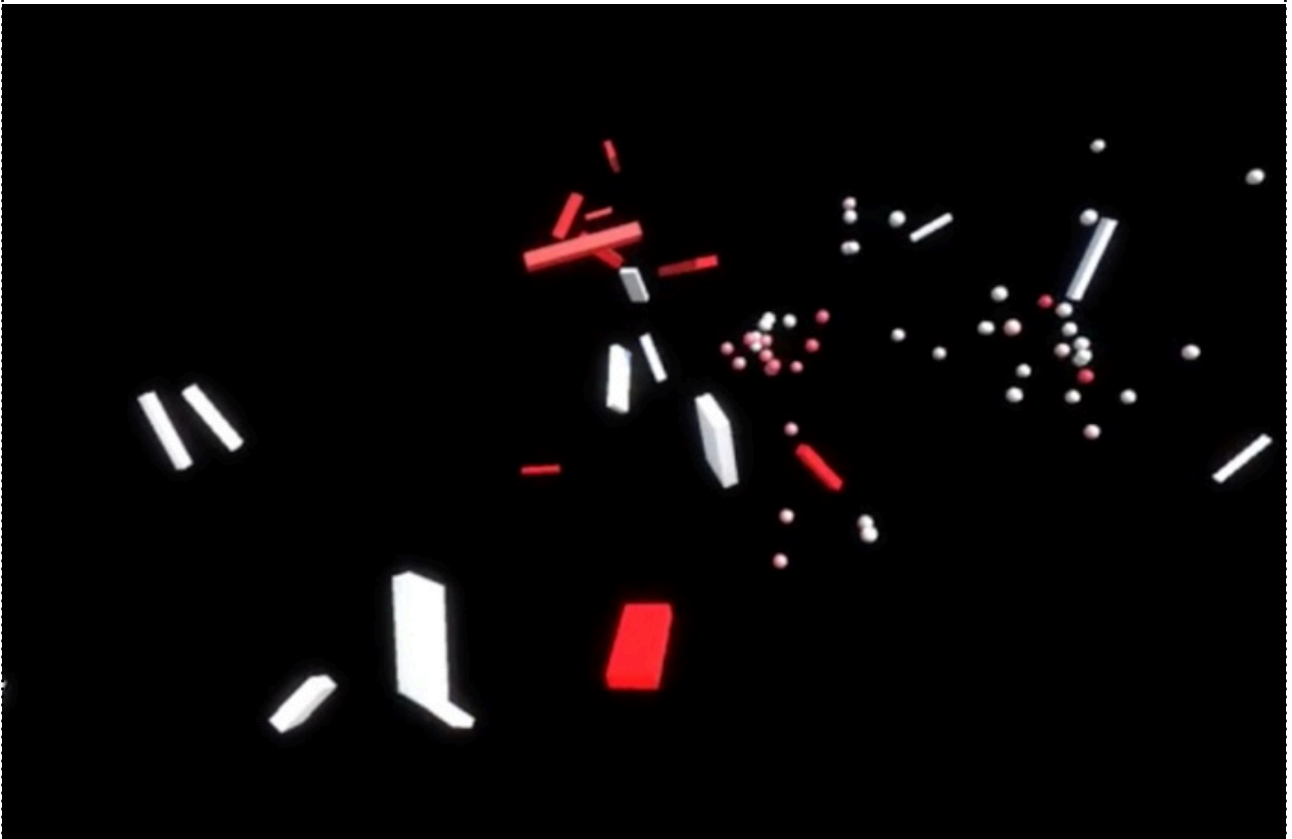
1.3.4 Gravity

A fundamental difference between how 65daysofstatic, with our four piece band template, and a laptop soloist approach the question of representing liveness in electronic music is that the solo performer cannot so easily be tied to the sound that is emerging from the speakers in the same way as when wielding acoustic instruments. One popular option for live electronic music is to shift the focus towards visuals of some description. *Gravity* follows this path and is a rough prototype built as an answer to the question: in what ways might physics-based composition offer a compelling approach to live electronic music composition? It uses the 3D physics of the Unity game engine to simulate gravitational pulls and the collisions it triggers to compose music with. It was built specifically to be presented in the context of being one of three performances

1. Phase One

included in a lecture based on looking at different approaches to live electronic music, and was designed to be used as a comparison between a short piece of live coding using the TidalCycles language (looked at more in section 1.4) and a rendition of *Full Bleed* from the *Full Bleed A/V* show seen in section 1.2.

Gravity Demo

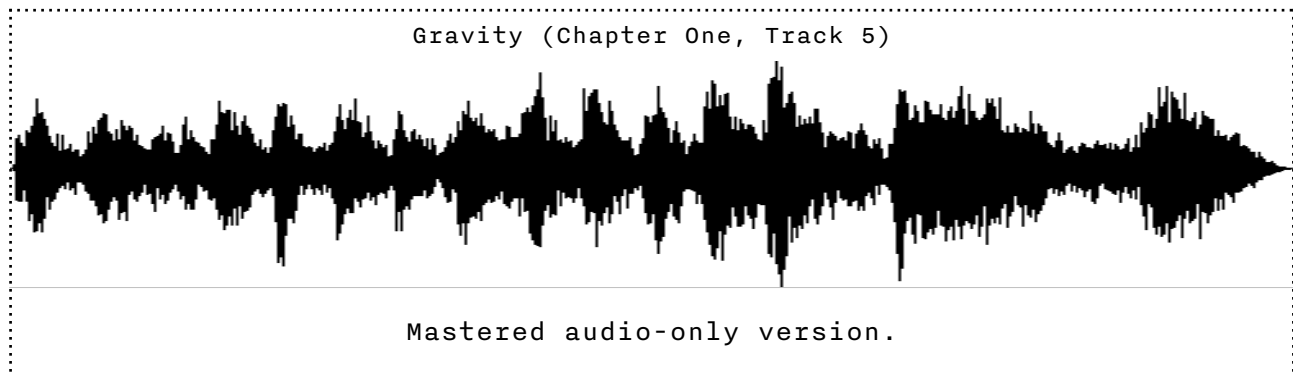


Screen capture of the performance.

Video Title: VIDEO_5_GravityDemo

YouTube link: <https://youtu.be/KTflX1EDHgQ>

1. Phase One



Built in Unity, *Gravity* focused on four different Gravity Spheres (GS). The full commands that the performer can issue are as follows:

- Spawn a sphere that takes GS 1,2,3 or 4 as its centre of gravity.
- Switch all active spheres to have either GS 1,2,3 or 4 as its centre of gravity.
- Revert all spheres to their original centre of gravity.
- Remove a random selection of spheres.
- Generate a piano phrase.
- Generate a celesta phrase.

The 3D environment is built in Unity. The performer triggers these events within this space by pressing the appropriate keys on the laptop keyboard. These events in turn spawn MIDI instructions which are passed to Ableton Live to generate the audio. Spheres can be created with a gravitational pull toward one of four different coordinates. The gravitational pull for the spheres can be changed in real time. When the spheres collide, they produce a note. The pitch of the note is based on the coordinates of the sphere at the moment of collision.

The rectangles are generated live by a script I wrote called Phrase Maker running in Unity. This generator is an algorithm that produces six note melodies in a variety of rhythms from a fixed pool of notes. The size and shape of the rectangles reflects the notes chosen by the algorithm.

1.3.5 Reflections

Gravity is a confluence of several emerging strands of thought, still early in the overall arc of this research. The original question of how physics-based manipulation of a sound space might affect composition is in some ways eclipsed by other considerations that emerged in its development. The Phrase Maker in particular allows for a more direct injection of my musical ideas and produces a more appealing kind of generative music than the harnessing of 3D space. Although relatively simple, the next section looks at this algorithm in closer detail as it is the first attempt to automate my own compositional preferences. Without any formal training in computer programming and prior to becoming more familiar with both the specifics of the C# programming language and fundamental code design patterns in general, this phrase-making algorithm illuminates a stepping stone toward generative music algorithms and techniques used in later research.

Using the physics engine to drive the music, while interesting, did not produce musical material that felt strong enough to stand alone as a performance and its arrhythmical nature made it difficult to sync to other compositional elements. As can be seen in the video, ultimately it is used to create a sound similar to granular synthesis, which is effective, but also unavoidably leads to the conclusion that as far as the final musical production is concerned, using an actual granular synthesiser would lead to more flexible results. Secondly, in contrast to the other two songs performed during the lecture-recital it was built for, from a personal perspective *Gravity* was the least successful. In terms of the amount of input I had into the music, it sat in the middle. One of the other songs performed was from the *Full Bleed A/V* project in which I had very little influence over the sound, whilst the other was a short amount of live coding in which I was very directly involved in the sound being generated. When compared to *Full Bleed*, *Gravity* lacked the production values and a cohesive arrangement, based as it was on improvised

1. Phase One

triggers that instantiated generative events in 3D space, and compared to the live coding it lacked the means for me to effectively edit and curate the output in real time.

Out of the three performances, the live coding performance got the most engaged response from the audience, although it should be noted that this was a conference full of academic musicians of different disciplines, most of whom delivered their own lecture-recitals, and this environment was far removed from the archetypal nightclub or live music venue more likely to host live electronic music, so it is difficult to draw firm conclusions.

1.3.6 Phrase Maker

Phrase Maker is a small piece of logic that sits inside the code for the *Gravity* system.

When it is triggered, it produces a new phrase based on the parameters I designed it with.

These decisions for the rhythmical part of the phrase maker came from noticing a pattern that emerged through improvising melodies on a piano to a loop running in 5/8. It seemed that notes had a strong tendency to be either one or three crochets in length. Wanting to try and build generative phrases based on this idea, I decided that each phrase should last two bars and, to give it variety, each bar should have the potential to be given a different rhythm.

```
public void Setup()
{
    phraseA = new int[] {1,1,3};
    phraseB = new int[] {1,1,3};
    reshuffle (phraseA);
    reshuffle (phraseB);
}
public void Reshuffle(int[] phrase)
{
    for (int t = 0; t < phrase.Length; t++ )
    {
        int tmp = phrase[t];
        int r = Random.Range(t, phrase.Length);
        phrase[t] = phrase[r];
        phrase[r] = tmp;
    }
}
```

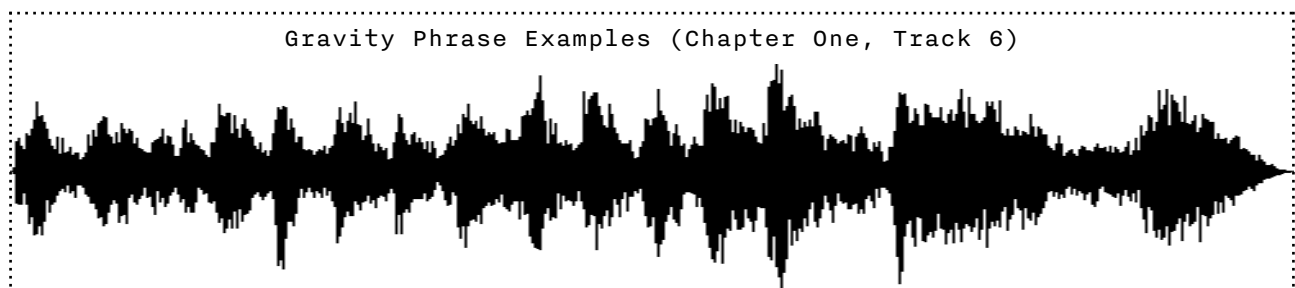
1. Phase One

Here is the very simple algorithm that generates the phrases. The first block here creates two simple arrays of numbers which will later be converted into lengths. In other words, each phrase will be made up of three notes: two notes that are a single crochet in length and one that is three crochets in length. The second block reshuffles each array. This is a very simple algorithm, but passing two bars through it and then combining them allows for 27 different rhythmical variations.

```
noteChoices = new int[] {60,62,63,67,72,75,77,79,80,84};
```

Combined with these 27 possible rhythms, the generator has a pool of 10 notes from which to pick 6 each time. In the code above, the notes are represented as MIDI notes. 60 is C3, 62 is D3 and so on. This increases the number of possible phrases to 4,082,400 if notes are not allowed to repeat within a melody, and up to 27,000,000 if they are.

Examples of output from this generator can be heard in the research portfolio in the track *Gravity Phrase Examples*.



This initial foray into generative music introduces an important consideration which remains present throughout the rest of this research: even relatively simple algorithms can quickly produce large amounts of material and so when it comes to composition then the contextualising of these algorithms is just as important as their inner workings. The examples heard in the above audio are the results of simple algorithmic logic being fed my own rhythmical inclinations, my own choice of possible notes and the output sculpted

1. Phase One

through my choice of instrumentation and sound design; feeding different aesthetic choices to the same internal logic could result in entirely different music. The algorithm is not a neutral entity, the choices it makes to create a phrase from two shuffled bars from an array of possible note lengths is as much my subjective creative choice as any other aspect of the music, but nevertheless, in and of itself it is as useless in a vacuum as a single note. It is only in its relation to other musical events that it comes alive.

1.3.7 Conclusions

The potential of manipulating 3D space within computer game engines to compose music seems like an area ripe for further study, in particular as a collaborative effort with games developers or programmers. However, the results of *Gravity* demanded a reappraisal of the scope of this research. This is a PhD in music composition, not computer music, and not computer game development. A unique contribution will not come from me teaching myself coding and engaging in forms that are new only to me, but rather in the experience of how these new technologies are intersecting the world of popular music composition. What am I, as a member of an active band and solo artist existing in the music industry, able to draw from these tools in order to make new work? In light of this, it was the generative potential and applied logic of game engines that emerged as the loci subsequent research focused on, rather than the specific creation and manipulation of 3D spaces.

1.4 Live Coding

1.4.1 The Lure of the Code

The move away from investing the time to develop complex 3D landscapes that could form and manipulate sound is a pragmatic decision to sharpen the focus of this research, but the concept of having visuals drive the music rather than music driving the visuals remains a compelling area of study because of its potential to open up new forms of creative expression in the context of live performances as well as the way it reveals previously invisible compositional patterns. Although in classical music the text of the work is visual in the sense that it exists as a score, in popular music the text is usually the sound recording itself. There may often also be partial visual representations of them in the form of music videos or live and recorded performances, but these texts are often approached as pieces of art in their own right rather than literal representations of the music.

In the practice of live coding, the performer's screen is generally projected behind them so the audience can see what they are typing. The text they type follows the syntax of any number of live coding languages and it is executed in real time, generating musical patterns. Much academic discussion surrounds this field, and plenty of writing has already been dedicated to charting its history and development (Collins et al, 2004), live performances (Magnusson, 2014 & 2016) and the visuals associated with live coding, including the aesthetics of the code itself (Cox et al, 2001), the spectacle (or lack of) provided by code (Roberts, 2016) and the ways in which audiences might perceive code during live coding performances (Zmölnig, 2016).

Like many of the areas explored in this thesis, my engagement with the community that identifies itself as live coders or live coding practitioners was somewhat ethnographical, with a focus on observing how the community worked, and then

1. Phase One

gathering useful techniques and recontextualising them in terms of how they influence my own creative productions, rather than trying to write an overview or critically analyse the way live coding as a movement might evolve as a musical form or contribute to popular music composition as a whole. As the *Zombie Coding* project in a later chapter will illustrate, my attempt to engage with a discipline that has deep ties to academia on its own terms provided a major breakthrough in my own approach to thinking about composition.

Leading on from the other Phase One projects, the main enquiries carried forward were in regard to how live coding could provide a visual element to my live shows, and how its approach to composition might be incorporated into and enhance my own creative practice.

1.4.2 YrCodelsBadAndYouShouldFeelBad

65daysofstatic released a song called *YrCodelsBadAndYouShouldFeelBad* as a piece of code on 20th June 2014. The full text can be found in Appendix III. This code was written in Gibber, a language developed by Charlie Roberts. At the time, the majority of live coding languages required, at best, specific software to be installed on a computer to be able to run it or, at worst, a significant level of technical skill such as to be able navigate a computer command line interface and use websites like Github to download and compile the environment in which the code would work. Gibber was different in that it ran in a web browser and was therefore easily shareable to a much larger audience regardless of their technical skills. This is a primary reason for 65daysofstatic choosing it, even though it was actually Alex McLean's TidalCycles language that I was becoming more familiar with.

The song was released via the band's Facebook post (65daysofstatic, 2014) with a link to the Gibber website (Gibber, 2014) that could execute the code in the user's

1. Phase One

browser and another to the Pastebin website where people could find the raw text (Pastebin, 2014). The level of response was so enthusiastic it immediately crashed the Gibber website. The code had a stated desire for listeners to “hack it to pieces” (2014). Some people evidently did experiment with manipulating the code (“Flip the p and z effects in ‘putOnYrSmilingMask’ and the difference is insane” reads one user comment (Facebook - 65propaganda, 2014)) but notably, within hours one fan had recorded the audio from their computer, re-uploaded the audio as a WMA file and linked to it on the Facebook thread discussing the release, the concrete musical form seemingly preferable to the open-ended, ambiguous nature of the browser experience.

The code itself was written with an eye to appeal to non-coders as well as people familiar with the coding language. This hierarchy of meanings is discussed by Zmölnig in *Audience Perception of Code* (2016), but in 2014 65daysofstatic’s approach was simply based on the values held by the band since their inception, which was the desire to share their music as widely as possible regardless of how interested a listener might be in the surrounding context of who made the music and how. Releasing a song as code wasn’t primarily done so people could see how we made music, because at that point we had never made any other music in this fashion, but to release music in a way that people could engage with as directly and simply as possible.

To this end, as well as being functioning code, the text also has some sense of an abstract, poetic narrative as comments woven into the musical instructions. Some excerpts:

```
//***** createInstrumentsFxDoubtAndUncertainty *****
...
// WhiteNoiseAnaesthetic
...
// coldSynthAnUndefinableSadnessCheerUpThoughEh?
...
//
switch_to_clean_beats_kill_the_cold_synth_bit_glitchy_but_then_isn'
t_everything_these_days?
_pretend_it's_a_metaphor_for_the_future_ok_ok
```

1. Phase One

(65daysofstatic, 2014)

This release took place shortly before this research began. At the time we were, as a band, experimenting with new ways of working. The *No Man's Sky* project was on the horizon but not completely confirmed and we were deep into the development of an installation called *Fugue State* (65daysofstatic, 2014) which was using code in a different way, based as it was on programming an Arduino to drive the magnets in six e-bows distributed across three guitars. Releasing a song via code in this context seemed like a natural extension of the way we were operating as a band during this period. The lure of the internet and the easy, instant open-source distribution of code was also looked upon as some small emancipation from our contractually-bound relationship with the commercial music industry. Releasing a song for free as code didn't technically break our record contract in the way releasing a piece of recorded music online would, and anything that might antagonise our record label or articulate our general disdain for the music industry is something that generally gets a green light from the members of 65daysofstatic. As the ASCII aesthetic borrowed from pirated nineties computer game splash screens implies, and the opening line of *YrCodeIsBad...* states: "noRightsReserved2014" (2014). If there was any declaration being made here, it was that as a band we wanted to find another way of doing things.

1.4.3 Going Live

Moving on to the specific live coding research contained here, the essential distinction between any previous experiments is the real time aspect that live coding performances demand. I was already familiar with improvising or jamming electronic music by manipulating or creating parts on the fly live. The Maschine software and controller for example, or Ableton's Push device, offer ways of composing electronic music quickly from basic ingredients. This sort of approach tends to be loop-based. Meanwhile

1. Phase One

trackers, less popular today than they were in the nineties, are still available and offer a different, less linear, less left-to-right approach to composition.

What distinguishes live coding with TidalCycles from the above is its conception of time. Although a cycle is locked into a time, the contents of that cycle are not divided by beats according to that time. A pattern with four elements will take the same amount of time to cycle as a pattern with five elements. This encourages the use of polymeters. Secondly, the concise nature of the code means that a lot of information can be displayed at once and is very quickly accessible. This is especially noticeable when manipulating melodies. Achieving the same effect using a controller like Maschine would mean that first the relevant group would be need to highlighted, the pads set into the correct mode and then, without a keyboard available to input precise instructions, notes would be shifted around using pads or knobs. With TidalCycles, it is as quick as pointing the cursor at the numbers to change and typing new ones. Furthermore, because so much data is visible at once, it is possible to change multiple values without updating the code and then execute those changes all at once, allowing for major changes to be made with a single key press. What is lost in return is the direct physical relationship with the sound changes that controllers can provide that is proportional and gestural, as well as some detail and control over single notes that dedicated DAWs and controllers can provide, given that for TidalCycles, the manipulation is generally happening at the higher level of rhythmic and melodic patterns.

All of this highlights a significant feature of live coding which is that the liquidity of the code means in the heat of a live performance notions of normal pop song structure can be difficult to maintain. Unless a copy of your original code block is made before making changes, a performance might quickly changed all the elements inside of it leaving no written record of what the song sounded like before. This lends itself to live

1. Phase One

coding performances taking on a somewhat episodic structure, albeit with very loose transitions making it impossible to define the point one episode ends and another begins.

The primary goal for me was to become familiar with live coding techniques in order to apply them to my own research, which is not live coding focused, looking at ways to expand creative expression in the context of my own creative practice. To this end, the practice-based research using TidalCycles that is submitted here comes in two flavours, the first of which is a captured live coding rehearsal. This rehearsal was recorded live, but not streamed live. The syntax is TidalCycles which is sending patterns to Dirt, a software sampler that accompanied early installations of TidalCycles. (At the time of writing the default sampler is now Superdirt, which runs inside SuperCollider).

Live Coding Rehearsal

```
slow 2 $ every 3 (density ) $ sound (samples "glock{3,8}" (irand 16)) |> pan (sine1) |> speed "1 3"

stack [sound "piano:2 piano:4 piano:5 piano:1?" |> gain "0.8" |> pan "0", sound (samples "piano*4"
silence

slow 2 $ stack [sound "tmp ~ ~ tmp ~ ~ ~ ~", sound "hh*8" |> pan (sine1) |> gain "1.1", sound "hc
sound "tmp [hc hc?] [tmp, sn:2] hc" |> shape "0.9"

slow 4 $ sound "pad" |> speed "0.5" |> gain "0.84"
silence

chop 64 $ stack [sound (samples "dnb*8" (irand 16)), sound (samples "amencutup*8" (irand 16)) |> pe
silence

slow 2 $ sound "lorde:4 ~ ~ lorde:4*4" |> cut"2" |> shape "0.5" |> speed "2 1 1 2"

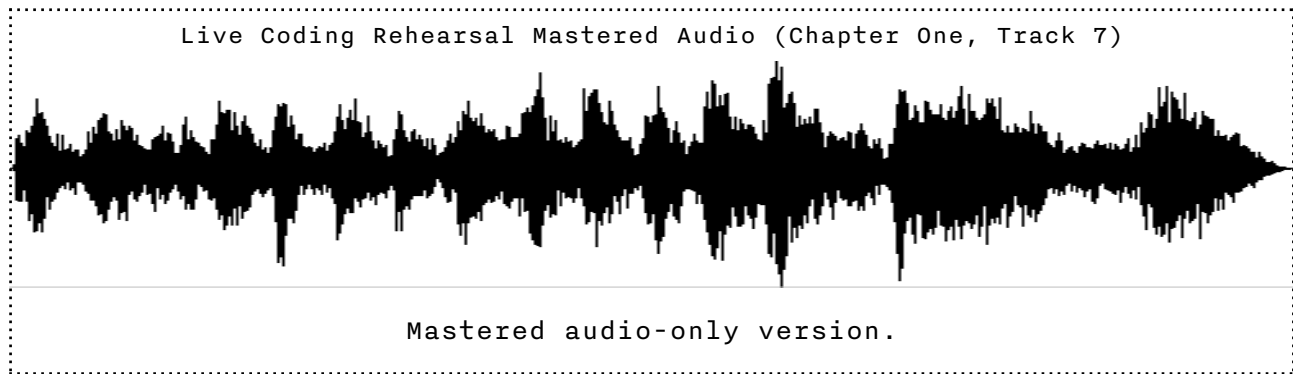
sound "bs ~ ~ bs ~ ~ ~ ~" |> shape "0.75" |> speed "1"
```

Please note the audio does not start immediately.

Video Title: VIDEO_6_LiveCodingRehearsal

YouTube link: <https://youtu.be/DFbNS7p-Xeg>

1. Phase One



My reflections on this piece are mixed and highlight the importance of context when making and presenting music. The performance and creation of the music itself using live coding was an enjoyable and satisfying process. Despite having worked with electronic music in various forms for more than a decade, the feeling of closeness to the music when writing it as code was remarkable and in the moments where I felt I had got a nice cycle running it became a satisfying mode of creative expression. Watching the recording afterwards, I found the video compelling in part because the code evolves slowly enough for me to be able to follow the changes. However, this slowness that helps make the video interesting makes the music itself unrewarding. Listening to it, it felt to me like an initial sketch at best, something that would require a lot of editing, production, and have more parts written for it, before getting anywhere close to being something that I would consider ready for release through traditional popular music distribution channels. This slowness in musical development, with parts within the music that I would choose to remove entirely, are inherent to the improvisational nature of live coding when it is actually performed in real time. The next step was therefore to look at how else these tools could be used in order to sidestep these limitations.

1.4.4 Invisible Coding

This second flavour of live coding focused on the non-linear, compositional potentials of the TidalCycle syntax without adding the complications of performance. This provided

1. Phase One

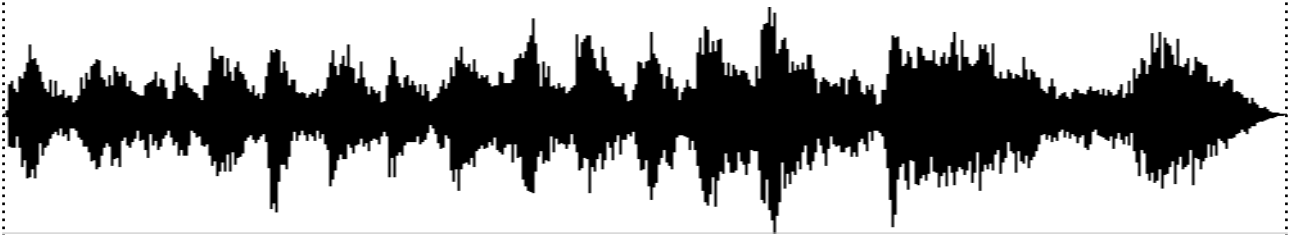
two significant differences for my composing. The first was that blocks of code could be written and refined over a period of time. (This approach could also be used for performances by preparing blocks ahead of time and pasting them into the text editor during a live performance. This is discussed in section 3.3). The second was that what is happening on the screen is not a concern. This meant that fast switching between the text editor that is driving the code and other music software to adjust other parameters would not interrupt a coherent, text-based visual aesthetic for the audience. The Live Coding Manifesto declares that "Obscurantism is dangerous. Show us your screens" (Toplap, 2010). If the whole process is shown to an audience, my own preferences when considering performing live with code are in line with the ideas discussed in 1.3.1 about 65daysofstatic's approach to building a live performance. That is to say, the aesthetics of a purely text-based mode of communicating live composition appeal to me over any notions of authenticity or transparency that are bound up in the live coding manifesto. As a performer, showing an aesthetically pleasing representation of process is preferred over revealing the process itself.

The main point here is that free from the pressures to perform or create in real time, a code-driven generative music system can be experimented with more freely with the intention of making standalone pieces of music.



1. Phase One

Tidal Tests (Excerpt #1) (Chapter One, Track 9)



Recorded live, excerpt from a TidalCycles improvisation.

Tidal Tests (Excerpt #2) (Chapter One, Track 10)



Recorded live, excerpt from a TidalCycles improvisation.

These three pieces of music are subjectively more successful than the real time rehearsal. “Flatland” was written at the request of Alex McLean for public release in the form of a mix of live-coded material for the MixMag website. (2017). This song has a higher standard of production and sound design than previous TidalCycles experiments, but still does not sound as well mixed as music I would put out through more traditional channels. The two Tidal Tests are edited out of a much longer performance. Heard out of that context, they succeed as experimental but dynamic and evolving pieces of music. This curation was essential in making them compelling, as when listened to in the middle of a 45 minute improvisational performance they were lost in derivative, arrhythmical glitches and miscellaneous audio cul-de-sacs. The production quality works only because of the glitchy aesthetic of the sound. It leans in to the rough and ready sound quality that TidalCycles provides.

1.4.5 Conclusion

For somebody coming from the programming world without necessarily being able to play an instrument or be familiar with more mainstream electronic music production techniques, creative coding opens up a completely new entry point into being able to make music. As a musician making electronic music for more than a decade before taking up live coding, the benefits are subtler, and come with drawbacks. The most compelling aspect of it in terms of applying it to my creative practice is the speed with which it is possible to make sample-based rhythmical patterns from disparate elements. For example, if I wanted a rhythm of kicks and hi-hats, this would be fast in most pieces of musical software. If I wanted to make a pattern that repeats the same rhythm over and over but chooses randomly each time from a thousand kick and hi hat samples, in TidalCycles it would be almost as fast, just a few extra characters to type. If I wanted to change a single kick drum to a roll of four kick drums over the same period of time it is as easy as changing 'bd' to 'bd*4'. I could change it to 'bd*256' if I wanted to trigger 256 kick drums in place of a single one and the result would probably be a very high-pitched glitch, possibly pleasant, possibly not, but either way this change would take about a second, whereas manually inputting 256 notes into most systems would take considerably longer.

The main drawback for me is built into this way of being able to create patterns from disparate samples so easily, which is that in terms of sound design and production, it can be inflexible compared to commercial music software. Whether in Ableton Live, Logic, Kontakt or ProTools, sampler instruments generally have the means to send individual sounds to separate outputs in isolation so they can be eq-ed, compressed, sent to further busses and easily mixed in relation to each other. TidalCycles, by design, is based on streams of audio. Uniquely made patterns combining many different

1. Phase One

instruments can be made extremely quickly, but all those samples are by default combined into a single stream of audio. It is technically possible to overcome this limitation and the live coding discussion boards and chat rooms are full of helpful advice in how to approach this, but for me this misses the point of what makes TidalCycles a tool distinct from other audio software. The rough-and-ready mixes are part of the liquid nature of live coding composition, so while the practice included here does not have a production value that I would feel comfortable releasing, they include rhythms that I stumbled upon through manipulating the code that I never would have thought to program manually.

Being on the periphery of this space is a very exciting place to be, as the live coding community is open, diverse, forward thinking and embraces the idea that music is as much about the audiences and culture it builds around itself as it is about any specific musical style or quality. It is a movement in which people identify as live coders in the same way somebody who plays a guitar might identify as a guitarist. For me though, while I can see that this is not necessarily a permanent state, live coding does not currently allow me to compose to the production standards expected in much contemporary electronic music, a world familiar to 65daysofstatic, in which every single element of a song is eq-ed, compressed, attention is paid to its place in the spectral domain of the song, a mix will involve many FX buses, sounds will be processed, re-recorded, re-processed, the balance of the overall mix being recalibrated with every change. To achieve this in live coding environments like TidalCycles currently requires the skills of a computer programmer rather than a composer. I do not doubt that in time this will change, but by the end of this research period I did not start to identify as a live coder. While recognising the technical achievements and ideas behind the TidalCycles language as a deliberately open music-making tool that people could apply in many ways to make all kinds of wildly different music, the focus of my thinking quickly moved from how live

1. Phase One

coding in itself might be incorporated into my own creative practice, to how I can design my own single-use, actively-biased algorithms to make precisely the kind of music I want to compose, instead of relying on this tool designed by the open source community for general use.

1.5 White Noise Festival

1.5.1 An Accidental Research Question

In December 2015 I headlined a small festival in Leicester. During this time, the music I was composing that I found most compelling was being generated through TidalCycles code. The idea of headlining a festival that featured many more guitars than laptops with an un-signposted live coding set was appealing, as it would force the music to stand up without any of the expectations and context normally found at live coding algoraves, the demographics of which “are distinct in relation to the relatively narrow range of professions they represent, which focus largely on roles related to, or involving, technology” (Burland & McLean, 2016).

However, with only a week before the show, the current iteration of TidalCycles in combination with my hardware revealed itself to be nowhere near stable enough to rely on for a headline slot. This left me with the pragmatic need of making sure I had a live show that I felt was strong enough to headline a festival bill, and also reliable enough to not crash the computer. These priorities took precedence over any conceptual or artistic desires I had. From direct experience of countless other live performances, the technical answer was to build a set in which the audio primarily came from Ableton Live rather than use TidalCycles or untested Max/MSP-based patches. This raised the question, what would be the most effective way of approximating live coding techniques in the stable structure of an Ableton Live project? The answer was provided in the form of the piece of practice-based research called *WNF*.

1.5.2 Pragmatic Strategies

To answer this question from 1.5.1, I looked at my performance strategies and the aesthetics of a live coding set that I most wanted to capture from my live coding performances to see how they could be replicated or approximated in Ableton Live.

These can be summarised as:

1. Perform in an environment where I can experiment with live manipulation of pre-made patterns and nevertheless retain an overall arc for the set.
2. Have some kind of visualisation of the musical patterns being generated.

Although the session view in Ableton is designed for live performance and improvised arrangements through triggering MIDI and audio loops in highly flexible manners, there are still limits on how easily changes can be made on-the-fly compared to TidalCycles techniques, especially when it comes to polymeters, which in live coding I regularly used to create variety and evolution in musical patterns. For example, in TidalCycles I would often have a pre-written melodic pattern that I would paste into the editor. For example:

```
d1 $ sound "piano(3,8)" # up "{0 3 2 5 7 3}%8"
```

This line plays a piano sample with a Euclidean timing of 3 pulses over 8 steps. ("X ~ ~ X ~ ~ X ~"). Following the structure of that pattern, the 'up' instruction changes the pitch of the piano note by adding an amount of semitones. Although there are 6 values in that pattern, the '%8' means that it moves through the pattern at 8 notes per cycle and since there are only 6 values, there are 3 loops of the 'up' pattern in the time it takes to do just 2 loops of the main rhythm.

1. Phase One

This example in itself is not particularly interesting, but demonstrates the possible complexity of rhythms in a very short line of TidalCycles code, and since patterns can be nested inside other patterns, a single line can create a rhythm built on logic and randomness that might never repeat itself in the exactly the same way twice.

When performing live, simply changing single numbers in a single line of code can have significant effects. Changing the Euclidean instruction (3,8) to (5,8) or even something like (<3 5 2>,8), which would iterate through (3,8), (5,8) and (2,8) rhythms over three cycles, requires only minor changes in values and it significantly alters the rhythms. To replicate this kind of behaviour in Ableton Live it would have required either a custom Max for Live patch that used the LiveAPI to dynamically create MIDI clips in real time, which was not an option given the time available for this particular project, or else literally building countless MIDI clips that mapped out a given phrase in a selection of possible polymeters. This too was not particularly viable.

1.5.3 MIDI Clip Wrangling

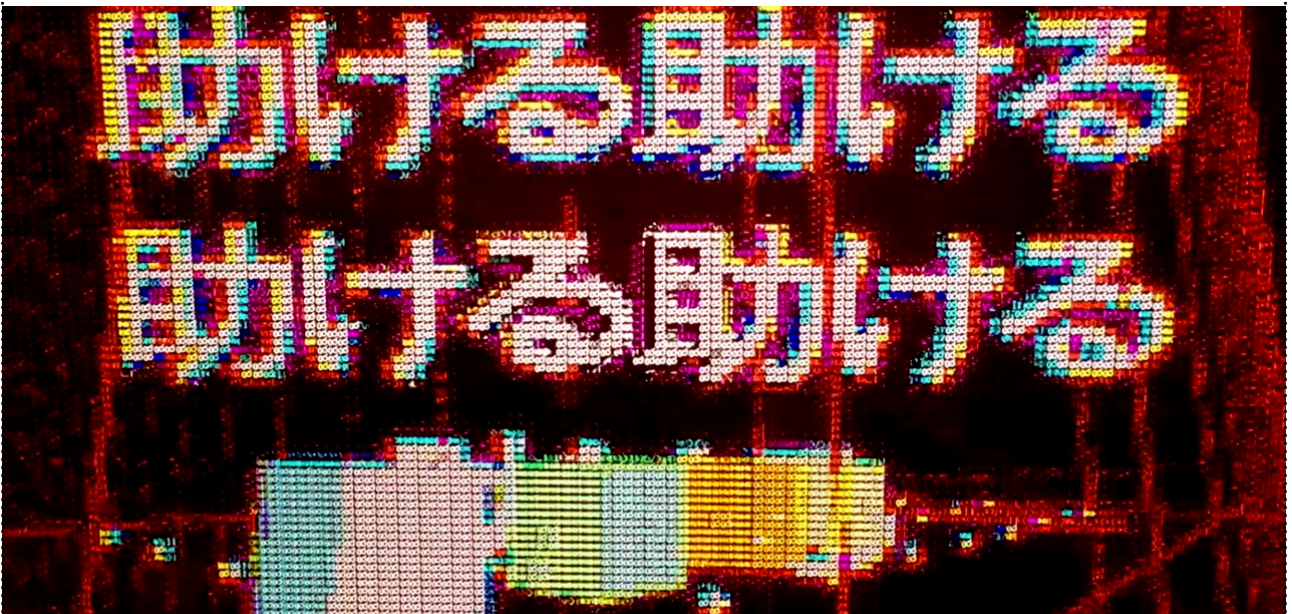
The compromise that was found used a specific feature built into Ableton Live's MIDI clips to get as much variation as possible out of a single clip, and to then make a number of clips for each part of each song that were variations of beats, bass lines, samples or melodies. The specific feature in Ableton Live is the 'Linked/Unlinked' toggle in the automation envelope box that sits inside every clip. By unlinking the automation loop from the MIDI loop, whilst still impossible to create polymeters in terms of triggering MIDI notes within a single clip, it is possible to create them within the automation. So for example, a drum pattern could be looped in 4/4 whilst automation on the transposition of the kick sample could be looping at 5/4, a filter cutoff automation on a hi-hat at 11/8, and the reverb send of a snare at 7/8. The beat remains in 4/4 throughout but the various

1. Phase One

automation curves run at different lengths, so at each repetition of the pattern the values are in different places, giving a simple rhythm the feeling of constant evolution. By layering up several MIDI clips horizontally, each with different time signatures, this can go some way to recreating the polymetrical possibilities so easily achieved in TidalCycles.

1.5.4 WNF.Output

White Noise Festival Screen Capture



Screen capture of visuals generated by a live performance.

Video Title: VIDEO_7_WhiteNoiseFestival

YouTube link: <https://youtu.be/pzW-vXkg3L4>

White Noise Festival Render (Chapter One, Track 11)



Mastered audio-only version.

1.5.5 WNF.Reflections

1. Phase One

The visuals for *WNF* were built in Unity using a similar system as the one employed in *Full Bleed A/V*. Inspired by the direct representation of notes and patterns in live coding, the entire set was based around a 3D grid. MIDI patterns that were playing the various drum kits and synths inside Ableton Live were also getting converted into OSC messages and being sent to Unity to spawn various 3D shapes or effects within this 3D grid, visualising the musical patterns live. The beats in particular were singled out, with kick drums triggering a script that glitched the screen whilst snares and hi hats instantiated objects on the grid. Given the time constraints, the aesthetic for the visuals leaned heavily on the 3D geometry and primitive shapes most readily available in Unity. They served a purpose, but much like *Full Bleed A/V*, remained some way from how I personally imagine my music.

Musically speaking, the show felt like a moderate success. The capture submitted here is 30 minutes long, whereas the actual show ran for one hour. It was weighted with new material in the first half, and from around 30 minutes new versions of old and known material made up the rest of the set.

1.5.6 WNF.Conclusions

This project was a success in that I built and successfully performed *WNF*, a show that I felt was strong enough to headline a small festival like White Noise in a short amount of time, thanks to efforts to integrate new technology into my compositional tools.

Significantly, the music itself was made using Ableton Live and some third-party plug-ins. There was no live coding, algorithms or generative Max patches, but my approach to existing technology had clearly been influenced by the non-linear, liquid musical structures that live coding in particular and the potentialities in generative music more generally had made me think about.

1.6 Music From Game Engines

1.6.1 Unity Triptych

The success of the Phrase Maker in the *Gravity* project and beginning to see the potential of algorithmic music through live coding moved me toward deeper research into how Unity could be used as a compositional tool. This next trio of musical experiments, *EGX*, *MONOSATT* and *Post-Keyhole* were conceived to both help become further familiar with programming and also to look at how the addition of a game engine to a suite of more mainstream music composition software might open up new possibilities for creative expression.

1.6.2 EGX.Project

A talk about the *No Man's Sky* soundtrack (see the next chapter) at EGX (Shrewsbury & Wolinski, 2016), a large video games conference, was used as a vehicle to test recreating one of the generative soundtracks made by 65daysofstatic within a 3D space, to see how Unity can be used to visualise a generative music system, and see whether this had any potential to aid the compositional process.

The 65daysofstatic *No Man's Sky* soundscapes were primarily put together in our studio using a piece of software called FMOD. At the time of building *EGX*, although FMOD soundscapes could be incorporated into a Unity scene, the FMOD API would not allow for the names of individual audio files to be retrieved as the sounds were playing. Because this was a requirement of the project in order to demonstrate how the soundtrack worked, the basic sound modules from FMOD were recreated within Unity. The primary function recreated from FMOD was a scatterer sound object. This is a type of module that can be filled with a pool of audio files, a time interval can be set with


1. Phase One

minimum and maximum values, and then the scatterer module will endlessly trigger a new sound at a different random interval within the given range.

This tech demo avoided BPM-based drum loops because, as standard, any scripting done in Unity runs on the main thread which is subject to fluctuating frame rates. Although this fluctuating means the timing may only vary by milliseconds, it can still cause the re-triggering of precise rhythmical loops to sound out of time. The level of programming required to code within a separate audio thread that runs independently to the game engine's frame rate was outside the scope of this particular research question, but is entirely possible to achieve otherwise.

1.6.3 EGX.Output

EGX Talk

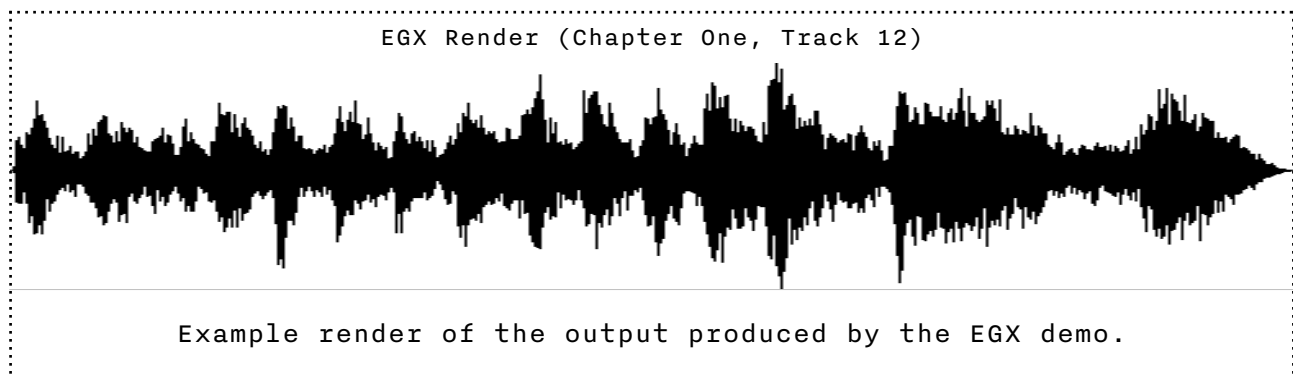


Footage of the talk itself which contextualises the software. The software is demoed from 10:20.

Video Title: VIDEO_8_EGXTalk

YouTube link: <https://youtu.be/bygXIGcSBMs>

1. Phase One



Anecdotal feedback gathered after the talk suggests that the demonstration succeeded in giving people a sense visually of how the soundscapes were being created. The aesthetic was deliberately impressionistic rather than focusing on trying to communicate too much data. The style borrowed from the sci-fi aesthetic found in *No Man's Sky*, the colour coding was set so that each instrument shared a colour, and the semi-random positioning of the audio files in 3D space was designed to be a visual equivalent of the way the scatterer sound functions.

Although visually appealing, the development process of setting up the various modules was more laborious than using FMOD. There is also a similar limitation to FMOD in the sense that Unity is not primarily designed for music, and although there is a complex audio mixer system within it, wanting to output multichannel audio, which would be a big concern when considering using a tool like this for a live performance or in a recording studio situation, would require additional coding.

While the visual system was useful in the very specific context of explaining how the mechanics of this generative music system worked to a general audience, it does not immediately lend itself to other contexts as a particularly visceral or engaging aesthetic.

Once again, most of the potential of using Unity to dictate sound rather than software like FMOD does not come from the visual element it can provide, but from being able to work inside a framework that allows for scripting to create musical logic. FMOD

1. Phase One

leans towards collected pools of audio that are triggered randomly, or that are arranged linearly but within atemporality, and instead of being moved through in time that movement is tied to variables which then get hooked into the game mechanics. The random scatterer modules are extremely powerful for sound design and tying the development of the audio to game mechanics without having to code is valuable when composing specifically for video games. But when considering how to use these tools primarily to make standalone pieces of music, the reliance on there being some outside agency to pull the ingredients together makes it less appealing as a compositional tool.

Whilst the audio-only version of *EGX* does its job as an atmospheric piece, it doesn't hang together spectrally or temporally the way a manually arranged composition would. However, unlike inside FMOD, within Unity there are endless ways to evolve the logic that is used to generate the composition.

Nested logic, behaviour trees, Markov chains and custom algorithms can all easily be added as code. An area of further study here would be to continue to use Unity as the driver of the composition, whilst offloading the digital synthesis and precise timing to dedicated audio software to allow for higher quality sound production at the end point, and more depth to the generative logic at the start point.

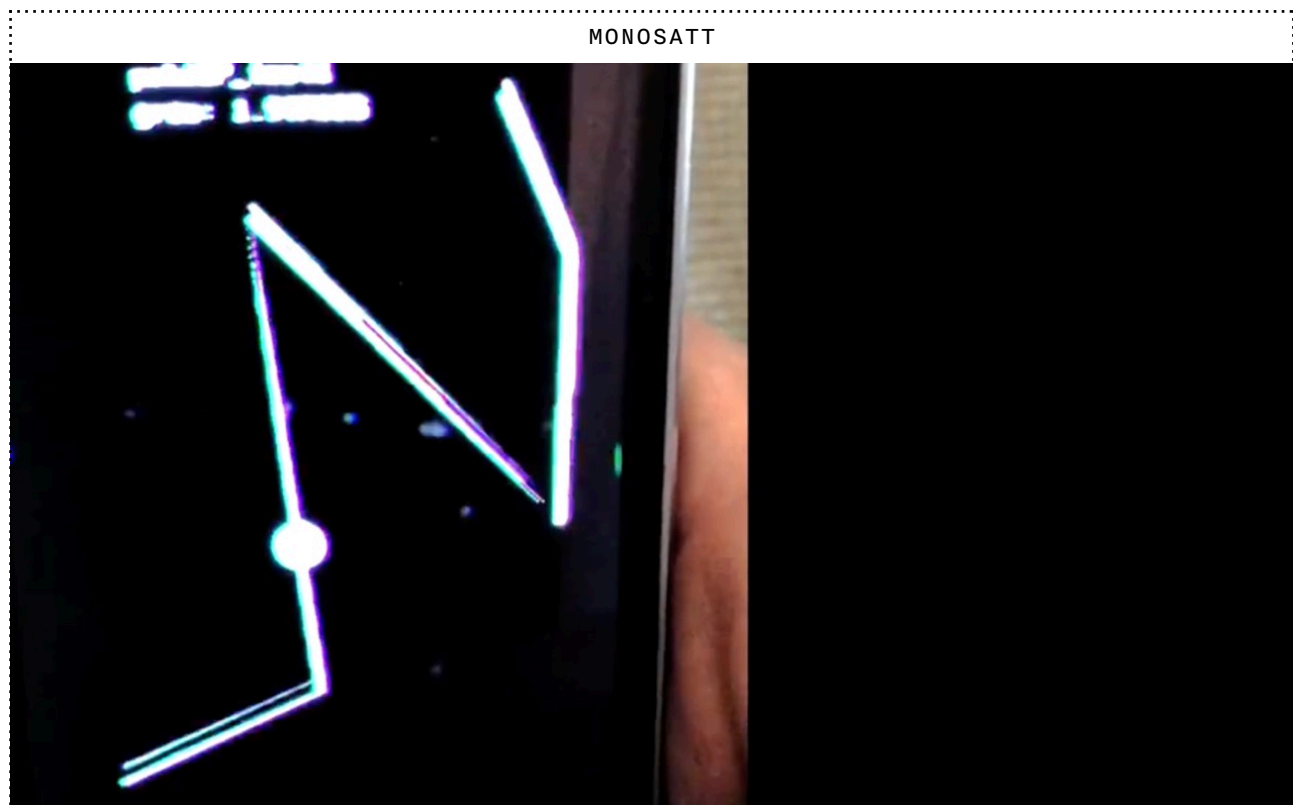
1.6.4 MONOSATT.Project

The conclusions taken from *EGX* suggested that to achieve a higher quality piece of standalone music, a dedicated combination of software running in concert should be the direction to take experiments in. The other two projects that ran more or less in parallel to the *EGX* development concerned themselves with also looking at new forms in which to present music that could only be achieved using game engine software. *MONOSATT* uses a very similar approach to *EGX* in terms of the technological approach. The generative

1. Phase One

audio is built in FMOD which then runs inside of Unity, which in turn drives the visuals in relation to both the user and the audio. The difference is, rather than this being a visualisation tool, it looks at the possibilities of presenting a piece of music as an iOS app.

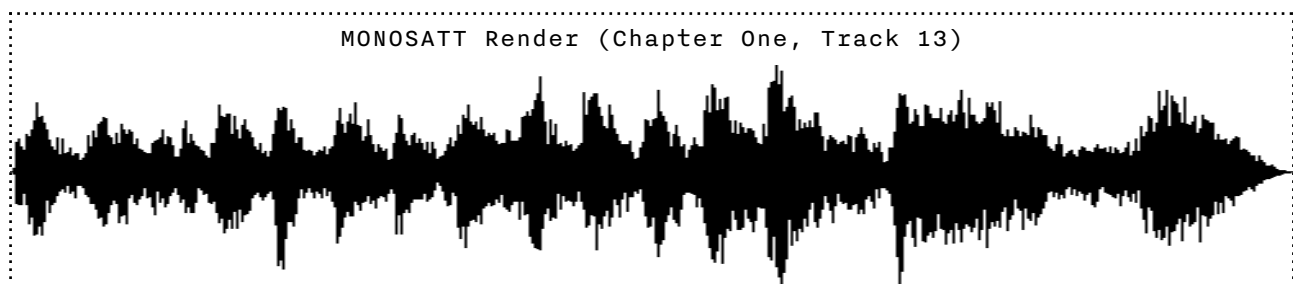
1.6.5 MONOSATT.Output



Demo footage of iOS app.

Video Title: VIDEO_9_MONOSATT

YouTube link: https://youtu.be/p_x85jd0XdM



Audio-only version.

1.6.6 MONOSATT.Conclusions

The FMOD logic is good at creating interesting combinations of sounds if the collections of sounds themselves are carefully curated ahead of time, but feels limited beyond that. For example, there is currently no scope to easily structure chord progressions or generate melodies.

From the perspective of a user, it is noteworthy how quickly this prototype stops being interesting. There are many mobile apps that are fully fledged music tools, and there are also many mobile apps for generative music, several with big names behind them, Brian Eno (2017) and Radiohead (2015) to name just two. Radiohead's *Polyfauna* (2015) takes advantage of the form by framing the generative music as a soundtrack to a game-like, 3D landscape the user can explore. Brian Eno's *Reflection* (2017), meanwhile, provides basic, unobtrusive visualisation to accompany the ambient music it generates.

Both Eno and Radiohead have a cultural currency and commercial success which allows for collaboration with specialists in this field and certainly, given the endless possibilities available in the computer game space and the speed with which that is developing, there is a huge area ripe for exploration there. My research here though is more concerned with how this technology can be used at the point of musical composition, from the perspective of a musician engaged directly with the technology rather than a programmer or interdisciplinary collaborator. On that level, given the same musical ingredients that make up this app, I feel I could have composed a more compelling piece of music manually.

1.6.7 Post-Keyhole.Project

The last piece of work in this collection was born out of a research question firmly rooted in the realpolitik of creating a live performance in the context of relentless commercial pressures. The performance was *No Man's Sky Live* by 65daysofstatic at the Sonar+D Festival in Barcelona (2016). The show itself and surrounding issues will be covered in the next chapter. What is noteworthy here is our decision that the best way to showcase a generative soundtrack was with a carefully time-lined, linear, non-generative performance. To further complicate the preparations for this show, Joe (65daysofstatic's guitarist) would have only recently emerged from keyhole surgery on his shoulder. Normally we would have simply not played until he had recovered, but the marketing push organised by Playstation around *No Man's Sky* meant that cancelling Sonar would not be an easy option for us.

The combination of all these pressures led to this unusual question: what generative music techniques could be employed to build a *No Man's Sky* instrument that is playable one-handed, but more engaging than typical effects achievable within a single knob or fader movement?

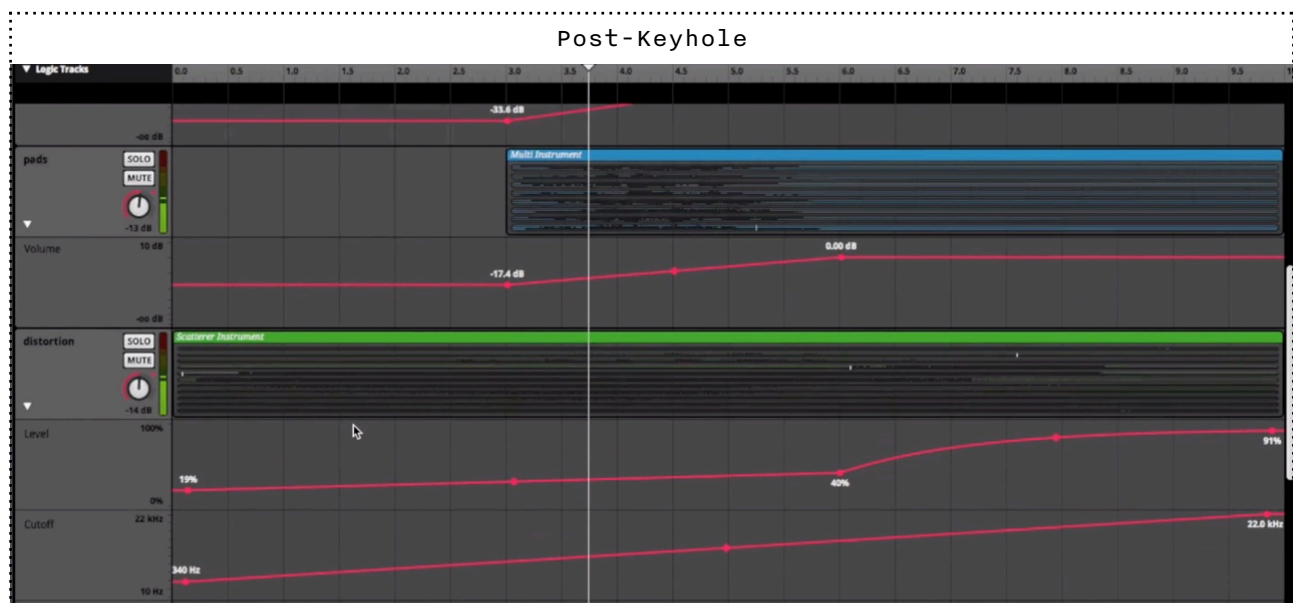
One-knob or single-knob plug-ins are common in both software and hardware incarnations. A single, large knob that does one thing well is a useful tool in live performances where it is often dark and hectic and precise, granular control is less often needed. A common use of them is to apply easily noticeable effects like filters, overdrives or reverbs. The aim of *Post-Keyhole*, the practice-based research made to answer this question, was more ambitious: to build a musical tool in which turning a single dial could manifest whole structures of noise.

A combination of FMOD with a simple Unity frontend was chosen for this. At this point in the *No Man's Sky* project we had a wealth of audio resources and sound design

1. Phase One

to pull from. The video below shows the FMOD project running live to catch a performance of the song. Although at first glance it looks like the familiar left-to-right timeline which is the foundation of countless DAWs, the audio is not in fact ordered in a linear timeline. Instead each clip is actually a collection of a number of pieces of audio and each clip follows different rules. For example in the video, the top row named 'DulciTone' shows 6 clips that actually each hold the same collection of audio files. These are called scatterer instruments, the same modules that were recreated in Unity for the *EGX* project. As the single knob at the top is slowly turned up, the song builds itself, adjusting FX parameters and introducing new blocks of sound which in themselves have simple logic attached to them. For the DulciTone track, although each clip holds the same audio, they each have decreasing time intervals between which new audio plays, so the further the knob is turned, the faster audio is triggered from this track.

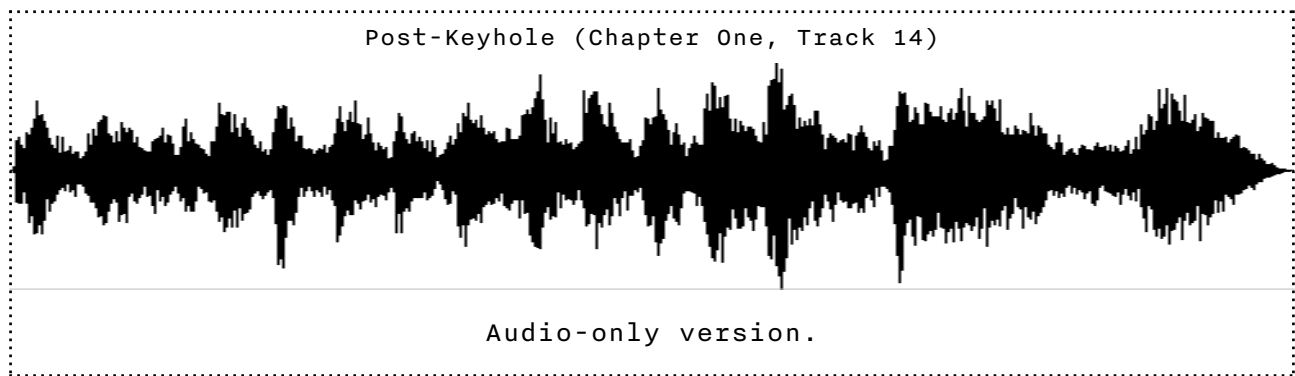
1.6.8 Post-Keyhole.Output



Footage of a live capture of the FMOD software.

Video Title: VIDEO10_PostKeyhole

YouTube link: <https://youtu.be/Fk4mG1DxnMU>



1.6.9 Post-Keyhole.Conclusions

Musically I feel *Post-Keyhole* makes for the most compelling listening of these three game engine-based projects, but it was ultimately never used. When 65daysofstatic tried to incorporate this into what needed to be a band performance with a large A/V element, this mode of one-handed performance became redundant. The generative nature of the piece, performed by one band member, would make it difficult for others to contribute, whereas a fixed arrangement would allow for all four of us to perform this song in a more animated way, with distinct musical elements that could then be mixed more successfully than the stereo output of the *Post-Keyhole* app could. On top of that, disconnecting this mechanic from any kind of visual, there would be no way for an audience to connect the sight of one band member turning a dial to the real, generative output being created.

This choice to abandon an otherwise compelling piece of music shows the potential dissonance between an academic environment of experimentation and the pressures of the commercial world of popular music.

1.7 Chapter Summary

This opening phase of practice-based research was designed to introduce new tools into my composing and push them in a number of directions at once. It also marked a significant shift in my working practice in one key way. Prior to embarking on this study, a huge amount of both solo and 65daysofstatic projects have, over the years, been unceremoniously scrapped. Countless hard drives are littered with abandoned, unfinished projects that will never be heard. This is because at some moment in their composition, these pieces were judged by either myself or the band to not have the potential to be good enough in terms of finished musical productions to be shared with an audience. In contrast to that rather merciless approach, everything undertaken in this thesis as practice-based research was kept, documented, contextualised and analysed in relation to the specific research question or questions it was designed to answer. Chapter summaries like this consider them in relation to each other and how they feed into the overall thrust of the thesis. As nothing is left out, failures, in the sense that certain pieces of research might produce music I was not happy with, were anticipated. These projects would still be important in helping to clarify which directions my research should take, and would often be successes when measured by other criteria despite them being musically uninteresting.

The first project, *Full Bleed A/V*, sidestepped this problem by being based on music that already existed. While I found the 3D geometry aesthetic to be of only limited success in terms of representing the mood I felt was embedded in the music itself, the software framework that I put together provided a key technical bedrock for everything that followed. Unity was reappropriated from its primary mode as a video games engine and repurposed as a malleable tool to generate visuals that could be very directly tied to

1. Phase One

music in realtime, while Max came to play a vital role in being a conduit between Unity and Ableton Live, where the audio side of the performance was based.

Gravity played an important role in following up this initial groundwork in a few ways. Firstly, *Gravity* existed in the context of being a piece of research made for my *If In Doubt, Make it Loud* lecture-recital. This marked the first time in the study a practice-based research approach was used to compose new music specifically to answer research questions. In this instance, it was the question raised by *Full Bleed A/V* about how the manipulation of the virtual 3D spaces enabled by Unity might lead to new ways of composing music, and the question raised by the lecture-recital itself which looked at how this technology might reveal new modes of performance for a laptop soloist wanting to make live electronic music.

Secondly, it helped me distinguish this research-based approach to composition from my pre-academic practice. Given the specific methodology I had employed to create *Gravity*, it became quickly apparent that I would not be able to use it to compose music I felt was good according to my own standards of composition.

Thirdly, as a follow up to *Full Bleed A/V*, it helped further clarify the importance and utility of using Unity as a musical tool. *Gravity*'s Phrase Maker element using scripted logic showed that Unity could not only generate visuals tied to music, but could be used in a compositional role itself. At the same time, it became clear that code-based, utilitarian modes of using Unity were more appropriate to this research than exploring its 3D spaces, although using Unity's physics engine and virtual environments for composition is an area that could provide further research opportunities.

With these boundaries defined, the next piece of research looked at live coding, specifically the TidalCycles live coding language. Experiments with TidalCycles answered questions about how I could use live coding in my own practice, producing pieces of music like *Flatland*, *Tidal Tests (Excerpt #1)* and *Tidal Tests (Excerpt #2)*. It showed how

1. Phase One

applying it to live performance could open up new approaches to a more episodic, liquid mode of live composition (explored further in 3.3 - Zombie Coding). Building on the overall research in this chapter, it also opened up paths to thinking about composition in less linear as well as more programmatic ways. There are techniques and functions built-in to TidalCycles that needn't be exclusive to this software at all, such as the ability to generate musical patterns with a minimal set of high level instructions that can be manipulated in realtime by the composer, or the Euclidean algorithm that can be used to generate percussive rhythms. These are ultimately mathematical functions that I was able to recreate in my own software systems built in Unity, Max or some combination of the two, allowing me to make less generalised, highly specialised song or performance-specific software tools going forward.

The next project, *White Noise Festival* is the first example of this. At the time of the research, TidalCycles was limited in ways that made it difficult for me to be able to perform a headline A/V set using only this software. Instead, I built an approximation of how I had intended to use it in Ableton Live and Unity. This project I considered a success both musically and visually. Although the visuals were still some way from how I imagined the music should ideally be represented, they were a marked improvement from the *Full Bleed A/V* visuals in two important ways. The first was that, aesthetically, the general glitchiness and style I used in the design moved the overall look of the show away from the relatively generic 3D geometry seen in *Full Bleed A/V* toward something more personal. Secondly, inspired by the attitude of the live coding community regards the direct relationship between the visuals seen by the audience and the music that they are hearing, the graphics for *White Noise Festival* were tied as tightly as possible to the individual notes and rhythms that were being generated by MIDI clips in Ableton Live. In the overall context of this first phase of research, the main research question *White Noise Festival* was designed to answer was how successfully my various new techniques were

1. Phase One

able to function in an environment outside of academia, with all of its associated limitations and pressures. The answer showed that although there were limits to incorporating fast moving experimental software like TidalCycles into the practice of a working musician, there were plenty of options to expand on what is conventionally expected of a live laptop performer, even in the context of a festival environment where soundchecks and set-up or installation times might be constrained.

The final run of experiments in this first phase of research were based on a series of research questions designed to make Unity the central tool of composition and explore how it could open up opportunities to new forms of expression useful to composers in the context of popular music. *EGX*, like *White Noise Festival*, used Unity to visualise how music was being generated from quite a specific, technical perspective. It displayed the names of all the audio files currently playing as well as the playback position of each file. Furthermore, building on *White Noise Festival*, the audio itself was being generated from inside Unity, making it possible for the entire project to exist as a standalone audio-visual application.

EGX was followed up with *Monosatt*, which built on *EGX*'s findings by asking what would happen if the goal was to create a standalone app that functioned as a piece of interactive music. This led to the creation of an iOS app. The app showed that it was possible to successfully build single-purpose pieces of software that existed as a piece of music that could be interacted with by the listener, but it also demonstrated that basic interaction alone does not make for a compelling experience. Additionally, it highlighted a key limitation of this approach to generative music, which is that it very easily leads to a loss of intentionality in the composition as the composer hands over agency to the listener/user.

Finally, *Post-Keyhole* looked at creating a standalone app that functioned not as a piece of interactive music for a third party to use, but as a specialised tool for a band to

1. Phase One

use during a live performance. Technically and conceptually this project was a success. The piece of music it created for this research portfolio is a compelling listen produced to a high audio quality. Furthermore, *Post-Keyhole* exists as a piece of software that can produce endless variations of this piece of music, with a high level of control given to whoever performs with it. On the other hand, it brought the tensions between a practice-based research approach to composition and the more pragmatic needs of working musicians into sharp relief. Despite delivering a new, viable form of creative performance and expression to 65daysofstatic, it was rendered unusable by the specifics of our set at Sónar and the context of us being a 4-piece band, wanting to perform live. This tension between research and composition emerges more clearly in subsequent chapters, including the next one, in which the realities of composing within the commercial context of soundtracking a large video game are set against the continuing research questions being tested by this study.

2. No Man's Sky

2.1 Introduction to No Man's Sky

2.1.1 What is No Man's Sky?



No Man's Sky (Hello Games, 2016) is a space exploration video game initially released on the Playstation 4 and PC. Hello Games, the developers, created an “infinite” universe (Khatchadourian, 2015) that the player inhabits. There are 18,446,744,073,709,551,616 planets, every one of them unique. They achieved this using procedural generation. With this technique, instead of the entire universe needing to be stored as data, the computer generates the particular slice of the universe the player is in, moment-by-moment. Because of the persistent nature of the algorithms the game uses, this experience is consistent for every player in the world. Given the infinite nature of the game, Hello Games also wanted an infinite soundtrack. Initially, Sean Murray (Hello Games' director)

2. No Man's Sky

and Paul Weir (the game's audio director and sound designer), brought 65daysofstatic in to simply write fixed, linear songs for the game's soundtrack, with the song stems to be delivered to Weir who would convert them to procedurally-generated soundscapes, but by the end of the project 65daysofstatic's contribution to *No Man's Sky* was a soundtrack album that existed both as a piece of fixed media containing linear song and soundscape recordings available on traditional consumer formats (CD, vinyl, digital download), as well as around thirty custom-made procedural soundscapes that exist only in a generative, dynamic form, running inside the game. We also performed various live iterations of the material around the world and online, from TV shows in Las Vegas to a digital arts conference in Barcelona and rock clubs around Europe and Asia. This chapter examines the *No Man's Sky* project as a case study into the pressures, limitations and opportunities that generative music can provide in the context of popular music. It begins by giving a short overview of the whole project before going on to look more specifically at it from a compositional context (new approaches to composition were discovered alongside new complications) and a live performance context (how best to represent material that is supposed to be infinitely variable). It ends by reflecting on the ultimate successes and failures of the soundtrack, the whole game, and what areas of further research the project points toward.

2.1.2 Project Overview

65daysofstatic's involvement with *No Man's Sky* began in late November 2013 when Hello Games got in touch to request the use of the song "Debutante" from the album *We Were Exploding Anyway* (2010). The track was to be used on the game's launch trailer that was going to debut at the VGX awards in Los Angeles (VGX Awards, 2013). After further discussion, 65daysofstatic came onboard as composers in principle in February 2014,

2. No Man's Sky

although due to the game's development timeline, work didn't officially begin until September that year. Nevertheless, during the summer of 2014 a library of song sketches was created. These sketches were based on some concept art, discussions with Murray and the footage available in the launch trailer (Playstation, 2014). Throughout the preliminary discussions, Murray's preference was that the game's soundtrack should be the sound of band as they were, rather than the band actively trying to compose music in a different style to their usual output (S. Murray, personal communication, summer 2014).

In September, 65daysofstatic met with Murray and Weir to discuss the specifics. Creatively, we were given free rein and persuaded to approach the material as our next record, rather than lean toward any more generic science fiction soundtrack styles. A playlist provided by Murray showed a list of 65daysofstatic songs that had influenced the early development of the game. Put against the entirety of the 65daysofstatic catalogue, the songs chosen were of the more melodic output, less glitchy and experimental. "Debutante" (65daysofstatic, 2010) in particular, still essentially an instrumental track but one that used layered vocal samples relatively high in the mix, seemed to capture the mood of the vision Murray had for *No Man's Sky*. At this point the Pulse generative system Weir was designing for the soundtrack was still in development, so 65daysofstatic concentrated on writing fully structured, linear songs. The deadline for the writing was the end of 2014.

In December there was a promotional trip to Las Vegas for The Game Awards (2014) and also A Night Under No Man's Sky (Playstation Experience, 2014), which was a specially-commissioned live performance by 65daysofstatic at the Playstation Experience. Knowing this was on the horizon, we managed to finish and record demo versions of two songs: "Supermoon" (2016) and "Red Parallax" (2016) as not only did new material need to be performed live but a new gameplay video was also going to be launched online (Playstation, 2014) and it was to be soundtracked by "Supermoon".

2. No Man's Sky

The decision had been made to record the soundtrack in the same manner as the previous record *Wild Light* (65daysofstatic, 2013), so in January 2015 65daysofstatic spent two weeks at The Chapel, a remote studio in Lincolnshire, with their engineer and co-producer Dave Sanderson. In May 2015 we travelled to Scotland for two weeks to mix the record with Tony Doogan at Castle of Doom Studios. The album was finished at the end of May 2015 but by this point it was apparent the timeline for the game itself, originally conceived to be released in the autumn, had moved significantly. Not only was the game release looking to be pushed back to Christmas at the earliest but the Pulse system for the generative soundtrack was still not up and running. 65daysofstatic therefore started working on versions of the soundscapes that would ultimately become the second part of their album. With this set of soundscapes, the intention was to gather a library of audio ready for moving into the next phase of the project but also to experiment with structure. The first part of the album was relatively traditional in terms of the song arrangements and instruments. The soundscapes, by comparison, were ambient and built from audio components that lent themselves to be rearranged more flexibly. Whilst in the context of the game the goal of the soundscapes is often to be very understated and unobtrusive, we used this time to explore how to create soundscapes that would still work as a standalone listening experience on record. In doing so, it helped not only define the spectrum of possibilities when working with the generative systems, but also gave rise to useful questions about the essence of the various formal structures that were becoming relevant for the different contexts this music was ultimately going to be heard in.

The generative music work for Pulse began more rigorously at the beginning of 2016. Mastering of the final songs had been put on hold until it was certain no major editing or remixing would be requested from Hello Games but was now completed at Abbey Road by Frank Arkwright. The Pulse system was up and running inside of the

2. No Man's Sky

game and Paul Weir made regular trips to 65HQ in Sheffield to discuss the best way to make the soundscapes work. By this point, we had expressed enough enthusiasm to stay involved in the project of turning the soundtrack into a dynamic music system within the game in a direct way. The idea of simply handing over stems to Weir had evolved into a plan to atomise the recorded album material and build the game soundscapes from these more fundamental song components, as well as recording many more.

In April, with seventeen Pulse soundscapes delivered, attention turned back to the album release with artwork and release plans being finalised. By this point the official release date of the game was June 2016 and therefore the record was scheduled to be released at the same time. This date would ultimately slip to August. In the meantime 65daysofstatic performed a one-off set at Sonar Digital Festival in Barcelona (2016). This set contained unique versions of all the songs from the main *No Man's Sky* album, as given the flexible nature of the material the hope at this point had been that the eventual live tour might incorporate some of the generative, varied nature of the soundscapes within the game. This live performance of exclusively *No Man's Sky* material was used as a hastily-conceived experiment to explore that idea.

In August the game's reception was at best mixed and from some quarters the response was outright hostile. The level of hype *No Man's Sky* had generated up until this point was much higher than was usual for a game made by an independent games studio. This combined with Sony's marketing push on behalf of Hello Games meant that through pre-orders alone it was destined to be one of the biggest games of the year, and expectations amongst gamers were set so high they could not be met.

The result of this was that Hello Games continued to work hard on the game development long after release. From October to December 2016, 65daysofstatic toured Europe and Asia. Nominally a new album tour, given the mixed feelings within the band about the success of the Barcelona show and the low likelihood of Hello Games having

2. No Man's Sky

the time to get involved in collaborating in any kind of *No Man's Sky*-specific live A/V show similar to Barcelona and Las Vegas, the decision was made to let the new songs be folded in to the existing 65daysofstatic live show rather than build a performance focused primarily on the new material. What this meant in practice was that in any given live show of the tour, in a set of twelve to fourteen songs usually two or three would be *No Man's Sky* material with the rest coming from the 65daysofstatic back catalogue.

Hello Games continued to update *No Man's Sky*. In November 2016, they released the Foundation update which was favourably received. In January and February 2017, 65daysofstatic were commissioned to write another twelve soundscapes. Unlike the previous soundscapes, this batch could not rely on existing material to be pulled apart and were written from the ground up.

In March 2017, the Pathfinder update to *No Man's Sky* was released that included the new soundscapes, signalling the end of 65daysofstatic's involvement in the project. Around this time Playstation released a 65daysofstatic studio session of three *No Man's Sky* songs which had been recorded the previous summer. In this session, a unique version of "Red Parallax" driven by the TidalCycles live coding language was performed (Playstation Europe, 2017).

2.2 All That is Solid...

2.2.1 Variform

65daysofstatic undertook four distinct approaches to composition over the course of making the *No Man's Sky* soundtrack. These four different phases break down as follows and are expanded upon in the rest of this section.

1. **The Main Album.** This is made up of songs from the first part of the double album *No Man's Sky: Music For an Infinite Universe* (2016).
2. **The Album Soundscapes.** This refers to the manually constructed, non-generative music from the second part of the same album.
3. **The Pulse Soundscapes.** These soundscapes are the in-game, generative soundscapes that appeared in the video game at launch and which were based on material from the album.
4. **The Pathfinder Soundscapes.** This is the last batch of soundscapes made for the Pathfinder update. They are generative, in-game soundscapes, but unlike the Pulse soundscapes were composed from scratch, not based on material from the *No Man's Sky* album compositions.

2. No Man's Sky

65daysofstatic - No Man's Sky: Music for an Infinite Universe



Disc one in the main album. The second disc contains the manually constructed soundscapes.

Listen on Apple Music : [link](#)

Listen on Spotify : [link](#)

In-game Soundscapes (Chapter Two, Tracks 1-3)



This is a selection of audio recordings directly from the No Man's Sky video game, with the sound FX turned off. These feature elements from both the original audio provided for the Pulse soundtracks as well as audio from the Pathfinder update.

2.2.2 The Main Album

The initial deadline for delivering new material was around three months and during this time there was no question of working with any of the generative audio technology the game would ultimately be employing, so the first phase of composition followed a similar path as previous 65daysofstatic records, albeit with a more tangible inspiration. As discussed in numerous interviews, our standard approach to composition was complimented with an ongoing cataloguing of variations that a song might have.

One song has these two big guitar melodies that wrap around each other, and we've gone through so many variations of those melodies, as we would normally do," says Wolinski. "But we've kept those unused melodies, for use in this procedural, infinite music machine. When these elements are put into the game, and maybe used in a loop, they could become this ocean of guitars, depending on the dynamic of the game and how they're mixed together. (Diver, 2015)

This process of composition through iteration is how 65daysofstatic have always worked. Songs rarely appear in one go. They emerge slowly and mutate along the way. It is common for instruments that were fundamental to the identity of the song as it began to have been removed entirely from the final version, not because of their quality but because the intentionality we wanted to inject into the song changed. Once a version of the song was captured for a record, any previous iterations were jettisoned. This mode of composing worked well as a starting point for *No Man's Sky*. We still allowed the song to mutate as it was written, but we catalogued all the extraneous audio at the same time. This meant by the time we had written the twelve songs that make up the main *No Man's Sky* album each song also had a library of audio that included different takes, variations on melodic or rhythmic themes, and sometimes versions of the song built from entirely different sound palettes.

The finished recordings were mixed by Tony Doogan at Castle of Doom Studios. Apart from the archiving of sounds as we wrote, which was more of an added

administrative task than a new approach to composition, this first batch of material was composed in the same way as any previous 65daysofstatic record. It was not rigorously documented, and because this album was completed before we started engaging in any of the procedurally-focused techniques that would come later, it is relevant to the thrust of this thesis primarily in the sense of being the source material that we would feed into generative music processes.

2.2.3 The Album Soundscapes

Once the main record was made, we found ourselves in an unusual position. We had finished making what we viewed not as a commission or side project, but as the next canonical 65daysofstatic album. It was also a subset of a project we had no control over and that project's timeline had stretched significantly. (The album was completed in May 2015 and ultimately released in August 2016).

At this point we still had no definitive information about the Pulse system, but there were signs we would be able to contribute to it and so were eager to have material on hand that was more flexible than the archived audio from the main songs. Those songs had been written with big themes, dramatic chord progressions, and clear narrative arcs to fit a mood we felt was appropriate to the sci-fi aesthetic of the game. Although we didn't yet know what mechanics would be used by Pulse, we had the idea to compose material that deliberately avoided the same conventional structures, material that was more episodic or exploratory yet still with a recognisable identity. Here me and Joe from the band discussed how we approached this portion of the project:

[Interviewer]: There's often a tendency for generative audio to be quite ambient, sometimes to an almost dull degree, did you and Paul Weir consciously want to avoid that?

2. No Man's Sky

Paul: “Yes... basically ambience is a get-out clause, and you can hear it in computer games going back 20 years now. I’ve got vague memories going back 20 years now of playing Amiga games with generative music soundtracks that would be MIDI-based but very long, dawdling melodies and phrases. This soft, soundscapey thing that you couldn’t really hum and be more of a background presence.

“So that was something we were definitely trying to battle against, and Paul Weir had the same idea from the outset as well. So we were all pushing in that direction. But again, even just a single melodic phrase has to exist through a certain amount of time, even if it’s just a few seconds. You have to figure out fuzzy thresholds to see how far you can push songs into the ambient state of the game.”

[Interviewer]: How did the process come together step by step?

(...)

Paul: “The second part was the soundscapes. It’s interesting looking at it now, because the soundscapes on our record are not going to sound exactly like the soundscapes in the game will sound.

“We made them ourselves, and we made them quite linearly in the way they were put together so this was us trying to figure out how the soundscapes might work [in the game] before we had access to the actual in-game audio system that would create them. So it was almost like a thought experiment; it forced us to think of music in this other way, and at the same time create even more sound palettes and textures to figure out what might work.”

Joe: “The soundscapes are more minimal, less concerned with arrangement... ambient is the wrong word, because they’re more aggressive than that, but their form is atemporal. They’re more loop-based, they’re more meandering. That’s just given us an in to making that sort of music and it’s going to feed in to whatever we do next.

“I’m so proud of that whole body of work. The soundscape stuff is coming from a much more drone-based loop-based place, and I’m really looking forward to seeing how that feeds into 65 as a live band.” (Laing, 2016)

It is noteworthy that 65daysofstatic called the same material “linear”, “atemporal” and “loop-based” within the same explanation. It speaks to our deliberate disintegration of forms we were used to, forcing ourselves to look at our approach to composition through different lenses. The ambivalence toward ambient music was less a distaste for the form and more an articulation of the desire to avoid falling into that mode of composition due to technical limitations rather than creative choices. Brian Eno, well-known as a practitioner of ambient music, has described it as being,

2. No Man's Sky

...based on abstract expressionism: Instead of the picture being a structured perspective, where your eye is expected to go in certain directions, it's a field, and you wander sonically over the field. And it's a field that is deliberately devoid of personalities, because if there's a personality there, that's who you'll follow. So there's not somebody in that field leading you around; you find your own way. (Eno, 2017)

This deliberate lack of guidance, where the listener can find their own path through the music, lends itself well to interactive soundtracks in video games where the music responds to the actions of the player. The less the personality of the music is defined in terms of temporal structures like melody and rhythm, the faster it is able to adjust to what the player is doing. What 65daysofstatic were doing, ahead of knowing what generative music parameters they were able to work with, was trying to envisage a music that brought the listener to a similar atemporal landscape, but then moved through it with them, creating local temporalities within its endlessness as melodies and rhythms emerged and fell away again.

This soundscape material eventually found its form through more pragmatic means. By 2016, it was decided that the album would be released by Laced Records and they had agreed to release a special edition boxset version that included four vinyl records. The main album would be spread across two vinyl records and the extra soundscapes could be spread across two more. Laced Records, wanting to retain a high audio quality, asked that each side be no longer than twenty minutes. Whilst this involved a less than ideal weighting of songs for the main album, for the soundscapes we sculpted them to fit the form. For the CD release the soundscapes came on a second CD, divided into six tracks. This same division was how the soundscapes were presented for the digital release.

2.2.4 The Pulse Soundscapes

Video game soundtracks generally respond to input from a player to one degree or another, even if it's as simple as different music playing when they reach a new level of a game, or the music pausing when the game does. Beyond that there are various approaches to how soundtracks can be made more interactive. With rendered audio, the composer can maintain the same production quality as with any piece of recorded music and still be able to make the music somewhat interactive. This tends to be done with what Winifred Phillips calls “horizontal re-sequencing” (2014) in which pieces of music are split into discrete segments that can be rearranged in various musically pleasing ways, or “vertical layering” (2014), in which a number of audio tracks run simultaneously, but are not always audible. Unlike a multitrack recording a composer might make in a studio environment where all the layers are mixed together to achieve the desired output, in vertical layering the layers are more self-sufficient, meaning a base layer can contain an entire mix of elements and extra layers emerge additively depending on the player's actions.

If the music is not pre-rendered into finished mixes it opens up possibilities for complex compositional techniques to be embedded within a game. This can involve granular mixing and triggering of isolated audio tracks, individual instruments or loops running via MIDI, or custom audio engines. When Brian Eno worked on the generative soundtrack to *Spore* (Electronic Arts, 2008) it relied on libpd, a version of Pure Data, to drive the soundtrack. “The audio programmers in *Spore*'s team who worked with Eno, described creating adaptive/procedural music as “a different way to compose,” where you are actually “composing in probabilities” using game events to trigger musical variations.” (Plans & Morelli, 2012). This approach involves actual synthesis happening within the game itself, which means endless possibilities for song structure and melodic

2. No Man's Sky

content, but a somewhat reduced sound palette of instruments and sound design that can be created digitally rather than being able to rely on samples or recorded audio. It also takes an amount of computing power and resources that a game might not be able to budget for. A challenge for most game music composers is that they are working under technological constraints in regard to the amount of computing resources the game developers allocate to them. A telling quote from Winifred Phillips: “the game [I was composing the soundtrack to] would be distributed as a downloadable title from the Xbox Live Arcade service, so I knew that memory space limitations would prevent me from having too many simultaneous layers in my tracks” (Phillips, 2014)). What was unusual about *No Man's Sky* is that the procedurally-generated nature of the game itself meant the size of the game in terms of its assets was relatively small. Instead of having thousands of 3D models and textures sitting in a resources folder, everything seen in the game was generated by algorithms on-the-fly. This freed up a significant amount of processing power and disk space that could be dedicated to running the soundtrack and gave 65daysofstatic an amount of freedom that we took advantage of by applying a kind of brute force method to generating a large amount of audio.

Karen Collins suggests ten approaches to variability in game music when a level of control over the music is available:

1. Variable tempo
2. Variable pitch
3. Variable rhythm/meter
4. Variable volume/dynamics
5. Variable DSP/timbres
6. Variable melodies (algorithmic generation)
7. Variable harmony (chordal arrangements, key or mode)
8. Variable mixing

2. No Man's Sky

9. Variable form (open form)

10. Variable form (branching parameter-based music)

(Collins, 2008)

Working in isolation from the game development team, without direct access to the Pulse audio system that would actually drive the music, for these initial in-game soundscapes 65daysofstatic applied all ten of these approaches manually by taking each song from the main record along with the archives of extra audio we had collected in relation to them and building new libraries of audio. Some approaches to building these libraries included:

- Sending audio tracks from the original songs through amplifiers, FX pedals or outboard studio equipment then re-recording it with live manipulation of the effects. The same audio might be sent multiple times through different combinations of equipment, or simply with different approaches to manipulation.
- Building new sounds from synthesisers or guitars to create complimentary ambiances. These sounds had slow attacks and long releases that generally hovered around the root note of the sound, or cycled through the main chord progression of the song.
- Recording multiple takes of melodic phrases from the song, but with the emphasis on subtle variations to make each take unique. These recordings tended to be 10 - 20 seconds long.
- Record a large number of takes of single guitar notes with a long sustain. The note was the same each time but had small changes applied: the angling of the guitar created a different type of feedback, a delay time was changed on a guitar pedal, a microphone deliberately moved or amplifier settings changed.
- Guitars were sent through bass cabinets; bass guitars were sent through converted radio amplifiers; synthesisers were routed through guitar pedals.
- Custom Max patches were built based around song melodies. These patches would generate variations on melodic themes in the form of MIDI notes which would be sent to various hardware and software synths and re-recorded.
- Using the TidalCycles live coding language, song melodies were recreated in code and then manipulated in real time performances which were re-recorded.
- For soundscapes that included rhythm tracks, we recorded multiple takes of live drum variations as well as drum programming. These takes were then chopped up into standalone loops.

The 65daysofstatic studio was essentially converted into a giant patch bay that could route sounds through our entire collection of equipment in most ways imaginable.

2. No Man's Sky

Once we felt enough sounds had been collected to make a particular soundscape, a number of workflows were established to process them. These included:

- Sending all the individual sounds that had been collected through a specific chain of amplifiers or effects to give them a more cohesive aural quality.
- Batch processing techniques inside a computer using Adobe Audition to crop the sounds, add FX, compression, EQ and make sure every file was free of any clipping.

A known limitation was that a soundtrack made from combining pieces of isolated audio in real time would add complications to the mix. This was dealt with by choosing a consistency in volume rather than aiming for the kind of dynamic range we would mix our songs for in an album context, a decision made easier through understanding that our soundtrack was still just one part of the overall sound design for a game that would include a whole host of other sounds, including a quiet but constant underlying soundscape that more accurately fit Schafer's "acoustic environment" (Schafer, 1977) definition of the word than our score did, in the form of a series of audio pieces Paul Weir designed to represent the ecologies of the various types of planets featured in the game. Practically, this meant rendering every single piece of audio contained in a 65daysofstatic soundscape library to a peak level of -0.1db and then mixing them within the generative system itself.

At this point in the process, we would have a collection of audio files that we knew fit together to make a soundscape, but were still working in isolation from the main game development and had no direct access to Pulse. To work around this, we built our own generative music systems in FMOD using only the logic we knew would be replicable in Pulse. The parameters we needed to work within were:

- No more than six tracks simultaneously
- Each track could have its own volume, given a relative decibel level to other tracks. (The loudest track was set to 0db across all soundscapes).
- Within each track there could be a collection of audio files, which could be triggered either at random intervals within a range of time (measured in seconds) or

2. No Man's Sky

if the soundscape had a tempo, then sounds could be triggered every X number of bars.

- An arbitrary 'interest' level, between 0 - 100, could be applied to the tracks. For example, a track might only activate if the interest level reached 80 or above. We did not know during the production process exactly what this variable would be tied to within the game.
- Every track could be given a pitch range. Given the melodic nature of most of the soundscapes, this was rarely used except for occasionally doubling up certain drone-based tracks based on the same collections of audio, with one of the tracks pitched down 12 semitones to create a sub octave.

Building the soundscapes in this way, despite not having access to the Pulse system, allowed us to hear what the generative output was going to sound like. Once we were happy with each soundscape, all the audio would be collected, named and uploaded to a dropbox online storage folder. It would be accompanied with instructions for Paul Weir so he could recreate the conditions and combinations of sounds within the game itself. Each soundscape was made up of three harmonious but distinct sets of audio called Planet, Space and Wanted. The first would be what played when the player was on a planet, the second when they were in space, and Wanted was reserved for when the player entered into combat or an otherwise more urgent state. For this reason, only the Wanted subset of each soundscape featured any drums. An example of the instructions delivered along with the audio for each soundscape can be seen below.

****End of the World Sun Variant****

WANTED SOUNDSCAPE. 122bpm. 4/4

'Drums':

- eotws_wantedBeats_1 ... eotws_wantedBeats_42
- in all the time

'Drone':

- eotws_grainSpace_1 ... eotws_grainSpace_17
- in all the time
- triggered once between 12-45 seconds

'Melody':

- eotws_planetHiInterest_1 ... eotws_planetHiInterest_14
- in all the time

2. No Man's Sky

- triggered once every 16 beats

RELATIVE MIX LEVELS:

Drums: 0db
Drone: -12db
Melody: -20db

PLANET SOUNDSCAPE. tempo N/A.

'Cello':

- EOTWS_cello_1 ... EOTWS_cello_6
- in all the time
- triggered once between 7 - 11 seconds

'Bass':

- Mono_bowedBassGtr_Gmin_1 ... Mono_bowedBassGtr_Gmin_11
- in all the time
- needs to be pitched to -1
- triggered once between 22 - 50 seconds

'Melody':

- Same instrument as the 'Melody' from Wanted
- triggered once between 9-12 seconds
- high interest level

RELATIVE MIX LEVELS:

Cello: -8.5db
Bass: -22db
Melody: -23db

SPACE SOUNDSCAPE. tempo N/A

'Drone':

- Same as 'Drone' from Wanted
- triggered every 10 - 20 seconds
- in all the time

'Bass':

- EOTWS_reAmp_1 ... EOTWS_reAmp_15
- EOTWS_reAmpFX_1 ... EOTWS_reAmpFX_14
- EOTWS_reAmpLo_1 ... EOTWS_reAmp_9
- in all the time
- triggered every 21-30 seconds

'Guitar':

- EOTWS_gtrCleanNotes_1 ... EOTWS_gtrCleanNotes_32
- EOTWS_gtrCleanSeqLo_1 ... EOTWS_gtrCleanSeqLo_8
- for high interest

2. No Man's Sky

- either triggered between 0.75 - 1.5 seconds OR, perhaps a few instances with continuous volume, and the higher up the interest level you go, the faster they trigger?
- if that sounds good, the fastest trigger rate that still sounds good to us is about 0.5 - 0.75.

RELATIVE MIX LEVELS:

Drone: -24db

Bass: -35db

Guitar: -27db

2.2.5 The Pathfinder Soundscapes

Six months after the release of *No Man's Sky* Hello Games commissioned another batch of soundscapes. Although all the previous in-game soundscapes contained lots of new material, conceptually they had been deconstructed from either the main album songs or the album soundscapes. This time the soundscapes had to be built from the ground up.

To fill this gap, we returned to old, unfinished 65daysofstatic songs that had been abandoned not because we felt they lacked quality, but because we had not been able to find suitable arrangements for them. These pieces were loops, drones, or fragments of songs, music we thought had been consigned to the graveyard of old hard drives suddenly found a form in which to express itself. We applied the same production workflow to this audio, recording many variations on the main themes, switching the instrumentation, re-amping and effecting pieces of audio over and over again to create libraries with which we could create soundscapes.

2.3 No Man's Sky Live

2.3.1 What Happens in Vegas

In December 2014, well before the launch of the game, as part of its promotional strategy, 65daysofstatic were flown to Las Vegas for two performances. The first was a performance of the song “Supermoon” at *The Game Awards* (2014). This was the debut of that song and was timed to coincide with the release of a new trailer of gameplay footage, again soundtracked by “Supermoon”. It also marked the official announcement of our role as composers for the game’s soundtrack.

The Game Awards Live Performance

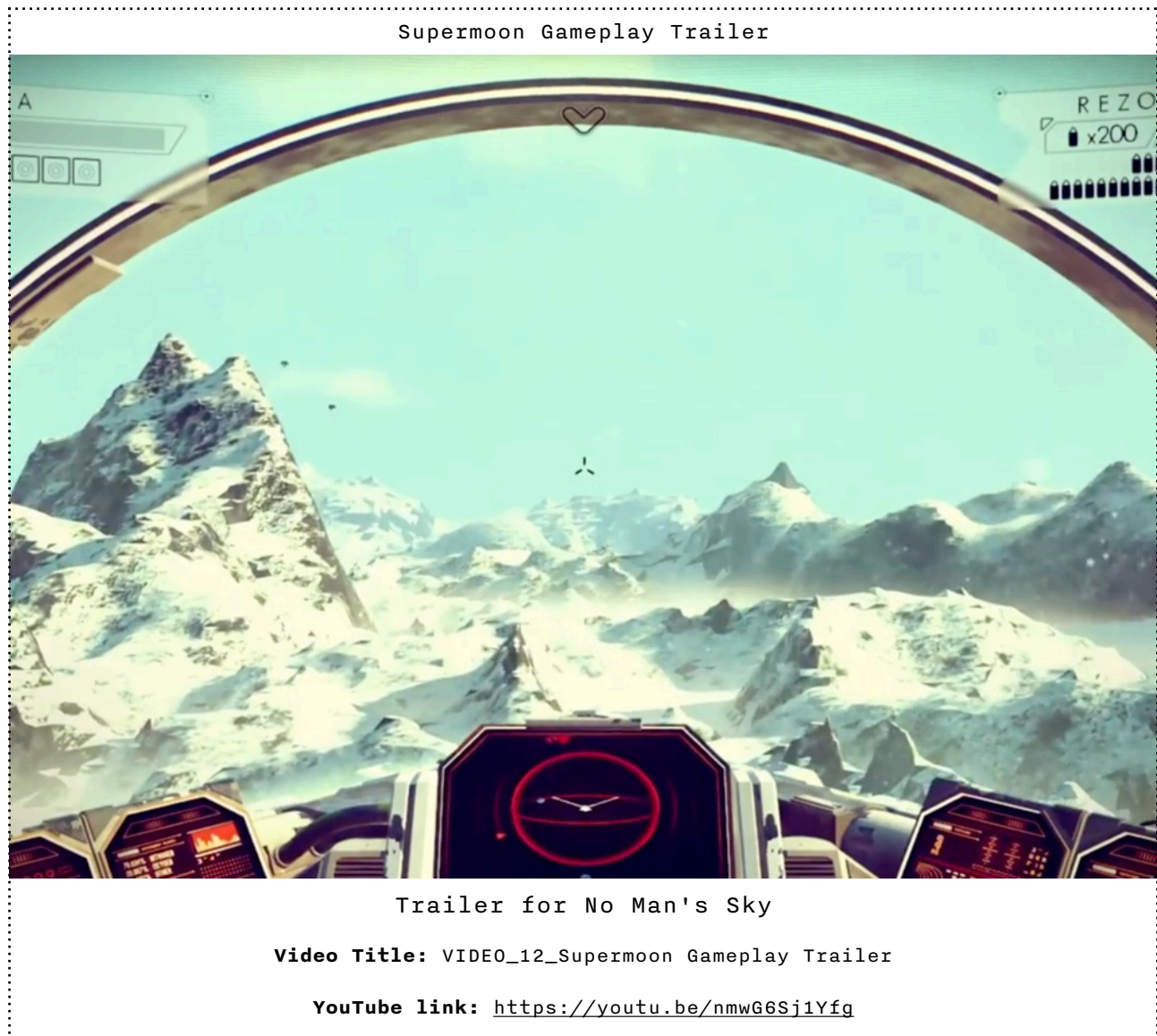


Footage of The Game Awards Performance.

Video Title: VIDEO_11_The Game Awards Live Performance

YouTube link: <https://youtu.be/Rt1o3skLcNI>

2. No Man's Sky



The following evening, still in Las Vegas, saw a second promotional event billed as *A Night Under No Man's Sky* (Playstation Experience, 2014) in which we played a full-length concert, streamed online, at the Playstation Experience, an annual conference in which gamers can come to try out yet-to-be-released video games. This concert featured two tracks from the *No Man's Sky* soundtrack, “Supermoon” and “Red Parallax” (65daysofstatic, 2016). The rest of the concert was made up from our back catalogue. Visuals were projected behind us throughout that displayed previously-unseen footage of the game.

2. No Man's Sky

A Night Under No Man's Sky



Full-length 65daysofstatic concert

Video Title: VIDEO_13_A Night Under No Man's Sky

YouTube link: https://youtu.be/BIXI11UY_s

Apart from the two new songs, the set was made up of older 65daysofstatic material which meant that aside from an uneasy feeling that our identity as a band could be easily subsumed by this game that was receiving more attention than we ever had, we were confident in the material and confident that we were able to put on a show to a high enough standard. The entire day leading up to the show was spent making sure that the visuals would run in sync to our live computers. During the show, it seemed that at least half the audience, not unreasonably, were more interested in the brand new computer game visuals that were being projected behind us than the show itself, and after the show we went out for fantastic cocktails with Hello Games. Being able to meet the whole game's team off duty and talk about things other than the game was easily the most valuable takeaway, creatively speaking, from the entire Las Vegas experience.

2.3.2 Sonar

2. No Man's Sky

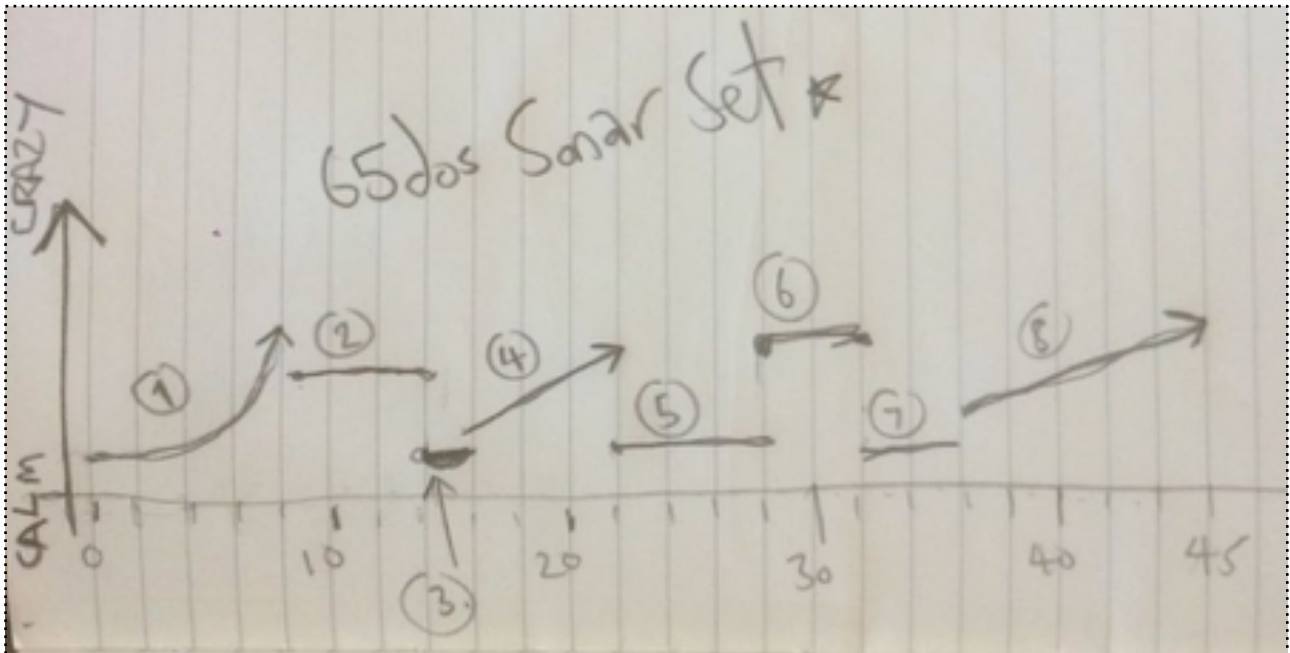
In June 2016 65daysofstatic were booked to play an A/V show at Sónar+D, the digital arts conference that runs alongside the Sónar music festival (2016). This show was to be entirely built from *No Man's Sky* material. As mentioned during the overview of the project, unlike the show for Playstation, by this point the bulk of the soundtrack writing had been completed and the material we composed existed in a flexible state within our FMOD systems, prepared audio stems, and vast libraries of isolated melodic, rhythmic and drone recordings. We had not yet established to what extent the touring around this record would be the typical 65daysofstatic approach of playing in live music venues, or might change into something that was exclusively or at least mostly comprised of *No Man's Sky* material that could be presented in some kind of generative or algorithmic form, perhaps developed in conjunction with specially made visuals from Hello Games, in theatres, galleries or planetariums. At the same time, the logistics of preparing the Sónar show were complicated by a lot of practicalities, not least our guitarist being post-surgery for a shoulder injury and unable to play the guitar. Additionally, there was no possibility of any kind of custom audio-visual system being developed and furthermore, only a very small amount of new footage from the game was available for us to make an hour long show that we were under no illusions was primarily a marketing push for the game.

The *Post-Keyhole* project from Phase One was a look at how we might be able to incorporate generative music techniques into this performance. Tools like this showed the potential for new generative compositions built from this wealth of audio, but did not lend themselves to a compelling performance. Combined with knowing that the visuals could not be directly tied to any generative process we might use onstage and knowing that the set had to be a strictly determined length, the question for us to answer became: what would be the best way to portray the scope of this soundtrack in a 45 minute live setting?

The answer we chose was to build the set entirely manually. The nature of algorithmic or generative music meant that employing it in real time would lead to

2. No Man's Sky

unpredictable results. Rather than risk a set of self-indulgent experimentation, we decided the most effective way of presenting the breadth of the soundtrack was by building a collage of songs and soundscapes that ebbed and flowed over the course of the 45 minutes in a linear, manually curated timeline. Here is a hastily drawn diagram that we sent to Zak Norman, the video artist we worked with whose responsibility it was to turn the few minutes of gameplay footage we had access to into a compelling 45 minutes of visuals, that shows the trajectory we intended for the set.



The show itself was not properly documented, due to miscommunication about it being filmed as part of the festival. Some footage from the crowd shows the Sonar version of “Supermoon” and audio of a recorded rehearsal shows the way that the setlist was built.

2. No Man's Sky

Sonar Live Performance



Excerpt of the show filmed from the audience.

Video Title: VIDEO_14_Sonar Live Performance

YouTube link: <https://youtu.be/UXGJi30ZZ98>

Sonar Rehearsal Recording (Chapter Two, Track 4)



Rehearsal room recording.

Given that we knew going into it that this show would be a one-off, it was not a relaxed performance. The compromises we made, especially concerning Joe's inability to play guitar, meant that all our choices in building the show were about making sure that the set had a carefully constructed arc. Rather than learning custom versions of all the songs, we each had our own cues, click tracks and notes. Knowing that Zak Norman was in charge of the visuals and that we would be unlit also helped in the sense that we didn't feel the pressure to perform with the same kind of physicality found in a usual 65daysofstatic show.

All that being said, the show itself was a success and if the release of the game had gone differently and there was scope for there to be a tour of a show built entirely around *No Man's Sky* material, it would have been a good foundation on which to create something even more compelling, perhaps with additional generative elements.

2.3.3 2Fly Studio Session

In August 2016 we did a live studio session that was originally intended for IGN, a large video game website, but was ultimately released on the Playstation YouTube Channel (Playstation Europe, 2017). The request for a live session had come from Laced Records but there were no specifics about what it needed to entail, so we decided to use this opportunity to finally apply live generative music techniques to a performance. The actual generative music tools we had used to build the soundscapes were based in FMOD and so not particularly performable. The question became: how can we use generative music techniques to drive the music whilst simultaneously making it clear to a general audience that this is what is happening?

In the event, it was a combination of TidalCycles, modular synths, FX chains and a system we had previously built that allowed for ebows to be controlled via MIDI. This can be seen most clearly in the first song, a live-coded version of “Red Parallax” (65daysofstatic, 2016).



This was a satisfying project as it gave us a reason to pull together various threads to make a performance that worked well in a studio environment. Live coding, plus our mechanical system for sending MIDI to ebows resting on guitars, are not well suited to easy integration with our existing live show. Being able to perform in a studio setting for later broadcast meant we could afford to take risks that we otherwise wouldn't be willing to. Ninety percent of the day in the studio was spent setting up and calibrating this system. The actual filming was squeezed into two hours at the end, where we did just one or two takes of each song.

Using live coding to rework non-generative tools felt like a success in that it represented the generative nature of the interactive soundtrack in an aesthetically more compelling way than showing the actual FMOD projects we used would have done, and it also felt rewarding to know that the *No Man's Sky* songs were flexible enough to be

2. No Man's Sky

reimagined for different forms in ways that found what was essential to those forms and made the most out of them.

2.3.4 Tour

From October to December 2016 we took our *No Man's Sky* record on tour around Europe and Asia. As a band on tour regularly playing songs from seven different albums composed over fifteen years with a wide range of technology and instruments, if we want to have any flexibility over our set from show to show there is a technical limitation to how we approach inserting new songs into the set. Unlike the previous promotional shows, this tour was booked from the perspective of showcasing a new 65daysofstatic album rather than as a chance to hear the *No Man's Sky* soundtrack performed live. To that end, rather than incorporate any algorithmic or generative music techniques into our setup, we instead chose seven songs from the *No Man's Sky* album: “Supermoon”, “Asimov”, “Heliosphere”, “Red Parallax”, “Hypersleep”, “Monolith” and “End of the World Sun” (65daysofstatic, 2016) and incorporated these into our live set in a similar fashion to all other songs.

Asimov & Supermoon Live in Hong Kong



Footage from Clockenflap festival, Hong Kong.

Video Title: VIDEO_16_Asimov and Supermoon Live in Hong Kong

YouTube link: <https://youtu.be/brB90L1W0iM>

Although these songs worked well in the context of our back catalogue, the possibilities this project raised about incorporating generative techniques into a live band performance we felt deserved further study. This is explored further in the *Decomposition Theory* project covered in Chapter Four.

2.3.5 Reflections on NMS Live

These different takes on presenting the *No Man's Sky* score live show that despite the numerous options we had to explore the wealth of material we made, manually-curated approaches made up the majority of performances. At both Sónar and at a video game event at the V&A Museum where we were asked to perform a DJ set of *No Man's Sky* material (65daysofstatic, 2015) we were given conditions that lent themselves to more experimental live generative music systems, but in both cases a planned linear arc of music was chosen as being able to deliver more compelling representations of the infinite variety contained within the score. The Game Awards showed a relatively muted performance of “Supermoon” constrained by the technical requirements laid down by the event producers and *A Night Under No Man's Sky* showed us performing hastily arranged, rough versions of songs from the soundtrack we were still in the process of writing inserted amongst the 65daysofstatic back catalogue for a performance in a context that suggested the audience would be more interested in the previously-unseen visuals from that game that were being shown behind us. The *2Fly Studio Session* was a deliberate effort to demonstrate a capability of performing with live, generative music systems, but involved a complicated, single-purpose technical set-up.

By the time the album was released and we embarked on a tour, any possibility of a radically reinvented approach to our live show based on the idea of performing with live generative systems had been subsumed by the decision that we were taking these songs

2. No Man's Sky

on tour framed as material from the latest 65daysofstatic album, rather than as the soundtrack to *No Man's Sky*.

2.4 Reflections on Infinity

2.4.1 Chronotypology

The *No Man's Sky* project made a lasting impact on 65daysofstatic and heavily informed this research despite many aspects of it being too loosely documented to be directly relevant here. It provided many opportunities to look at algorithmic music techniques from a compositional perspective, a live performance perspective and pointed to a interstitial form where as a band we were capable of producing compelling generative audio, but without necessarily having platforms on which to easily share it.

The nature of the video game form as a vehicle for music has opened up an area of study known as ludomusicology. It is an area that seeks to find a framework to analyse video game music on its own terms (Hart, 2014) as the context in which it is musical is tied to the agency of the listener in a complex way. This had led ludomusicologists to look at gameplay as a performative act of which the music is an expression of the player, and Darshana Jayemanne's "chronotypology" (2017) is a methodology designed to critique performances of video games by framing them as ensembles of diachronic and synchronic events. This framing allows for radically different performances to be compared. An example from Jayemanne is a level of a computer game separated by a cutscene. One player may take 40 minutes to reach the cut scene and then finish the remainder of the level in 20. Another player may rush through the beginning hitting the cutscene after 20 minutes, but then take their time finishing the level over the next 40 minutes. Although the performances of the two players might vary in countless ways, not only temporally, they can nevertheless be evaluated together using chronotypology. The cutscene diachronises the two halves of the level for each player, while also serving the synchronic function of bringing the two players together to have the same performative

experience regardless of either of their actions before or after. Two performances full of different combinations of action and distributions of time can nevertheless be described by the same structure.

Applying this thinking to performances of algorithmic music is an area for possible further study. It allows for structure to be placed on otherwise endless generative outputs without needing to temporally concretise that structure in advance. Using a chronotypological methodology to build a generative music framework points to a system where a composer could engage in a performance of improvised algorithmic curation for as long as they wish, knowing that when the moment is right they can switch to a musical ‘cutscene’ — a pre-composed piece of music as a synchronic event that diachronises the earlier algorithmic material. This idea of a diachronic-synchronic approach is explored somewhat in both *Zombie Coding* (Chapter Three) and *Decomposition Theory* (Chapter Four), but remains an area for future work.

2.4.2 Inevitably, it's Hegel

One of the main goals of the in-game *No Man's Sky* soundscapes was that they should be infinitely long. Conceptions of infinity can be traced back to Aristotle and can be seen woven into historical developments of philosophy and mathematics from then on (Nawar, 2015). It was Kant that first reasoned a solid connection between the two strands of metaphysical and mathematical infinitudes (Moore, 1990), but it was Hegel who drew a useful qualitative difference between true infinity and a spurious or “bad infinite” (Woods, & Grant, 2003) that can serve as a caution when thinking about compositional strategies for generative music composition.

Kant declared that the world as it really was in its metaphysical whole was infinite, but as human beings we can only receive it in finite moments of conditioned experience

2. No Man's Sky

as knowledge gathered through our senses (Moore, 1990). In order to recognise what we receive, this known finitude has to be set against the unknown infinitude it was plucked from. Receiving anything is bound up with an understanding that there is always something more behind it and this sets up the mathematical infinitude, an “infinite regress” (Moore, 1990), a quantised yet still infinite series of finite events. It is this mathematical conception that describes Hegel’s “bad infinite”. It can be thought of as an image of a straight line, stretching to infinity, there is always something more ahead and so this ‘bad infinity’ can never be comprehended all at once. In contrast the true infinite is the line “bent back upon itself, its image becomes the circle, the line that has reached itself, closed and wholly present, without beginning and end” (Hegel, 1969).

For Hegel, true infinity was beyond the grasp of human experience. We are finite beings after all. By this reckoning, endless streams of generative music are manifestations of bad infinity, no matter how intelligent the algorithms driving them might be, because as finite beings all we can ever do is tune into a musical stream that stretches off into the temporal horizon and hear sounds emerging as finite events.

Generative music doesn’t necessarily need to strive for infinity but in the context of *No Man’s Sky* that was precisely what was being attempted. Through the lens of Hegel the result, along with the game itself, can only ever be an example of the bad infinite, because neither the soundtrack or the universe can be fully experienced or properly comprehended in their entirety by the player. So what, in this context, would truly infinite music even look like? It is hard to imagine a stream of music bending back on itself to become a unified circle, an absolute, atemporal music that can be heard all at once. Humans can only experience sound through time and from a phenomenological perspective it can be said that “sound reveals time” (Ihde, 2007). In this context a true infinity could only be represented by a comprehensive understanding of a musical algorithm designed to play endless music itself, but only whilst it remains resolutely

2. No Man's Sky

unmusical. Reflecting on the music of *No Man's Sky* from this perspective raised one particular issue pertinent to this study: when is the illusion of infinity a better representation of what we understand by it than infinity itself?

The *Longplayer* installation (Finer, 1999) is currently in the process of playing a piece of music that will last a thousand years. The length is by no means infinite, but is many times greater than any lifetime and the fact that it has an end allows us to comprehend it in an abstract way that would not be possible with a mathematically infinite piece. Culturally, the length of a thousand years is fathomable. Our historical knowledge reaches back much further. Our imaginations can grasp some sense of the enormity of that amount of time passing.

The amount of raw audio recorded for *No Man's Sky* totals almost 30 hours and 65 days of static worked on the project intermittently for over two years. What could have been different if instead of writing music to feed generative algorithms that produced a mathematically infinite soundtrack, like *Longplayer* we focused on a finite amount of material and composed 30 hours of unique music that covered a wide sonic range of style, genre and mood? Or 20 hours of through-composed variations on a theme combined with a less intrusive, ambient generative music system?

These questions suggest a retreat to conventional musical structures might have been preferable (in this specific case at least) rather than the more experimental methods and forms that were used. The fact that these questions are being asked in relation to a study investigating possible new forms and algorithmic music techniques points the way to a tension between a practice-based research approach to new expressions for music and the priorities and pressures that popular music culture place upon composition. The emphasis on documenting and articulating a mode of production in practice-based research can result in different work to when the emphasis is placed on the final musical

2. No Man's Sky

production instead. This balance of process and production is a concern that is taken up in projects covered in the next chapter.

2.5 Chapter Summary

The *No Man's Sky* project was serendipitously timed to be able to incorporate it into this study. It was undertaken in an environment outside of the confines of this research and as such could not be documented to its fullest extent. At all stages of the composition, the moment to moment and day to day creative choices made by 65daysofstatic when writing the soundtrack were based on a reflexive, intuitive working practice rather than actively designed as practice-based research. Nevertheless, as a case study, and with my role as a member of the band giving direct access to the creative process, it illuminated areas in which new forms for creative expression can be further explored by musicians, offered various perspectives on their strengths and limitations and provided clearly defined methodologies to build upon going forward.

In terms of formal music structures, it showed that the same songs can exist in a number of states and that this can still make sense conceptually in the context of a band like 65daysofstatic. It demonstrated the kind of creative choices that went into sculpting the music appropriately to best fit its form and the freedom and potential it offered us in terms of our live performances, being able to make infinite amounts of thematically cohesive music. Looking back to the *EGX* project from the previous chapter, it also led to new ways of building musical systems that allowed us to distribute and present music in software form, tied into visuals.

Aside from the research questions that focussed on the compositional elements of *No Man's Sky*, from a cultural and commercial perspective it also revealed valuable information about the liminal space between practice-based, academic composition and being a musician working on a large commercial and technically complicated project.

In terms of technology, the generative music system for the game locked us into very strict parameters inside which we needed to make the soundscapes. Isolated from

2. No Man's Sky

the game developers, the musical systems that we built ourselves jumped between Max patches, Ableton Live Projects, software we made Unity and TidalCycles code, often all hooked up and running simultaneously across computers. These systems presented us with many more possibilities to create what we found to be compelling soundscapes. However, the amount of processing power and software frameworks involved in doing this were far beyond what Hello Games needed us to work within. Even if this level of complexity could be recreated inside their game engine, systems like this could still only fulfil their potential by being situated in our rehearsal room, with all the other amplifiers, instruments and recording equipment available to us.

This pointed toward further research questions: what is possible if we explore these kinds of musical systems outside of writing music for *No Man's Sky*? How could 65daysofstatic use them for live performance? How could they be utilised to write music not to serve as soundtrack material, but to exist in its own right? As shown in section 0.5, the latter period of work on *No Man's Sky* continued through most of Phase Two, which is covered in the next chapter. During that period, these new research questions thrown up through *No Man's Sky* were still gestating, to be explored more fully in Chapter Four and the *Decomposition Theory* project.

3. Phase Two

3.1 Introduction to Phase Two

3.1.1 Too Much Thinking

Engaging with new technologies and applying them to composition was not a new strategy undertaken for this thesis, it has been a fundamental part of 65daysofstatic since our inception when all the electronic programming for our early live shows came from an AKAI S2000 sampler and a Korg M1. Prior to this study, I had experience with: coded music (the aforementioned *YrCodelsBad...* (65daysofstatic, 2014)); generative music (the track "Piano Room" from my *Full Bleed* album (Paul Wolinski, 2014) is the captured output of automated MIDI generation in Max with some basic logic); as well as live performances and site-specific installations that dealt with generative visuals (*Sleepwalk City* (65daysofstatic, 2013)) and code-based mechanics to drive the music (*Fugue State* (65daysofstatic, 2014)).

What changed when embarking on this study was a new intention to formalise these efforts by paying more attention to the questions I was intuitively asking myself when doing so. Examples of this approach have been seen in the output of Phase One and the difficulties of maintaining a rigorous documentation of process in the domain of popular music can be seen in *No Man's Sky*. Parallel to this, a strain of theoretical ideas, whilst not directly tied to any one project, was shaping or perhaps more accurately loosening my opinions on what being a musician in this contemporary moment was and what it could or should be. The rest of this introduction provides a summary of these issues to help contextualise the thinking that shaped the projects created in Phase Two.

3.1.2 NaN Bpm

As mentioned at the beginning of this commentary, an initial aim that inspired this study was to address a feeling of atemporality in the context of making electronic music. These thoughts of a vaguely defined end-of-history moment in the music industry were subsumed by Fredric Jameson's writings on postmodernism. Jameson's essay *The End of Temporality* begins:

After the end of history, what? No further beginnings being foreseen, it can only be the end of something else. But modernism already ended some time ago and with it, presumably, time itself, as it was widely rumored that space was supposed to replace time in the general ontological scheme of things. (2003)

Jameson's postmodernism, which he also calls "the cultural logic of late capitalism" (1991), the globalised, instantaneous nature of the capitalist process has reduced its existence to a state of "constant present" (Jameson, 2003) and because the fundamental mechanics of capitalism require endless growth, if it can no longer grow through time then it needs to instead expand in space. In postmodern politics, this comes down to "essentially a matter of land grabs" (Jameson, 2003). Mark Fisher extrapolates from Jameson's ideas of postmodernism and late capitalism to posit a political-economic system he calls *Capitalist Realism* (2011), a state of existence "conditioning not only the production of culture but also the regulation of work and education, and acting as a kind of invisible barrier constraining thought and action". (Fisher, 2011). Popular music, as a subset of popular culture, emerges from and through this economic churn.

Addressing these ideas through the frame of this research project and its aim of discovering new forms for popular music practitioners to utilise, questions emerge: what does it mean to write songs within capitalist realism? What does it mean to write songs after the end of history? What happens when the emphasis is on putting music into space rather than time?

3. Phase Two

Thinking about atemporal music is not new. For example, Brian Eno approached this with soundscapes like his *Discreet Music* (1975), which was designed to be a “continuous, endless place in time”. The idea of music not as a “sonic narrative” but more a “sonic landscape”. “A landscape always in the present tense” (Eno, 2003). Such songs exist endlessly in some other place and the listener catches only a fragment, the form of the song is designed to give the impression of it being a window onto something much larger. Xenakis discussed the temporality of music, and even the multiple temporalities within a piece of music (Xenakis, 1992). Embracing the cultural impact of these ideas, record labels like Ghost Box and artists like The Caretaker have explored hauntology (another way of describing this feeling of atemporality) through their musical output (Fisher, 2013).

Being able to underpin the feeling discussed at the start of this study with solid foundations in critical thinking brought some clarity to how I viewed my creative practice in relation to socio-technical and economic forces at various scales, especially as I began to be able to critically analyse the extent to which capitalism shaped my compositional choices at a subconscious level despite my best intentions. Rather than this clarity guiding me to a space in which I could contextualise my practice, it instead led to a familiar question in regard to the inexorable machinations of capitalism on our lives: what is to be done?

3.1.3 Transcending Atemporality

It was at this point that the open-ended research of Phase One became channelled into some stronger themes. Theoretically, the desire to find a way of composing in or through postmodernism that didn’t lapse into nostalgia, a yearning for the past or an acceptance of an atemporal malaise, led instead to new considerations. Bourriard’s

3. Phase Two

“Altermodernism” (2009) and Jameson’s suggestion of the existence of “postmodern artistic singularity effects” (2003) are texts built around a convergence of multiple temporalities and spaces, where “utterly distinct and unrelated realities (...) [are brought] into relationship with each other”. (Jameson, 2003). Ideas about performatism were looked at as one way to “supplant the postmodern” (Rudrum, D., & Stavris, N. (Eds.), 2015). From a more utopian perspective, ideas from Attali about composition (Attali, 1985), Jameson’s “utopian impulse” (Jameson, 2005) and Srnicek and William’s politically charged call to “invent the future” (2015) were all taken onboard, and are discussed in more detail in section 3.2.1.

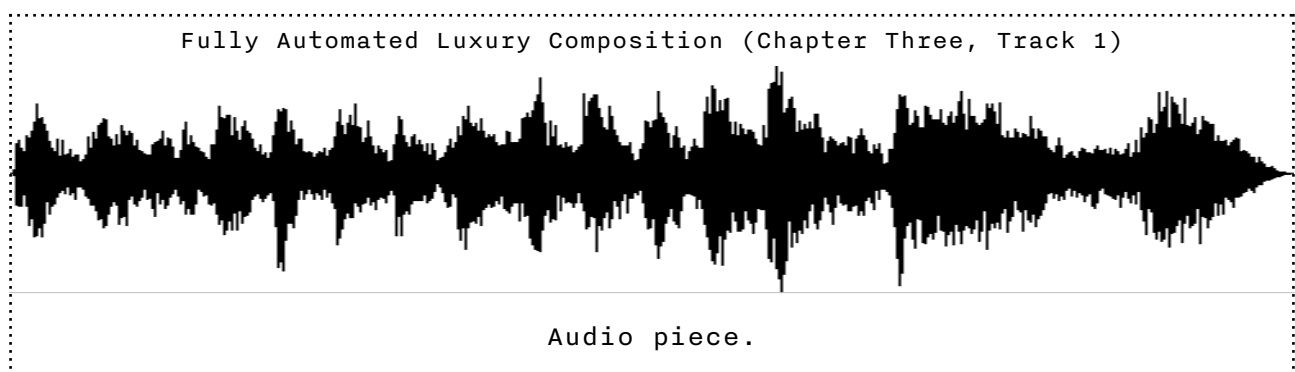
Practically, this meant incorporating algorithmic and generative techniques from the *No Man's Sky* project with the software frameworks developed in Phase One. Together they were focused into practice-based research questions grounded in specific ideas about the compositional process, whilst remaining sensitive to the emphasis that practitioners of popular music would place on the final musical production.

3.2 Fully Automated Luxury Composition

3.2.1 FALC in Context

Fully Automated Luxury Composition was published in IASPM Journal (Wolinski, 2017). Although submitted as practice-based research, it veers towards becoming practice-led research as defined by the distinction set out by Candy (2006) due to a heavy reliance on its theoretical framing. It looks at the question of how utopian thinking might be applied to popular music composition. The writing that accompanies the audio grounds the basis for asking this question in Jameson's ideas about the utopian impulse (2005), Attali's ideas about a future mode of composition in which anyone can produce endless streams of music (1985), and examines how this thinking resonates with Adorno's definition of popular music (1941).

Practically speaking, the question was answered by a compositional process in which I first built a series of generative music machines, recorded their output and then added additional melodies, rhythms and sound design around it to create a finished piece of music. The full published text that accompanied the audio piece can be found in Appendix IV.



3.2.2 FALC Reflections

Unlike the initial experiments of Phase One, and the arc of *No Man's Sky* project which was shaped by external forces, *Fully Automated Luxury Composition* was the first time generative music approaches in the context of popular music were put together so consciously and explicitly in this study. The commentary for the publication stayed focused on the idea of utopia in its central conclusion, but the work brought up areas for further study in regard to the tensions it revealed between the expectations of popular music, the infinite potentialities of endless music and capitalism's ability to subsume a potentially emancipatory practice like fully automated composition, alienating composers from their work to "make already-existing modes of creative production more efficient under capitalism" (Wolinski, 2017). Following this train of thought into the next project led to the existentially confused yet artistically valuable project *Zombie Coding*.

3.3 Zombie Coding

3.3.1 Zombie Coding in Context

Zombie Coding started as an artist statement about my approach to live coding as part of an aborted journal submission. It did not aim to be an academically rigorous, peer-reviewed piece of writing, instead the intention was to create a deliberately performatist work based on Eshelman's use of the term (2016) and explained in the next section, a piece of practice-based research in the form of a written statement and accompanying video. This work invented the concept of zombie coding, a technique to describe the act of pasting pre-written snippets of code and then manipulating them during a live coding performance, rather than typing out wholly new code in real time. This concept stemmed from the live coding experiments in section 1.4, and the ideas proposed in section 1.4.4 in particular.

As the artist statement acknowledges, this process in and of itself is not particularly groundbreaking, but that was not the point of this research. A performatist work is built upon a "double framing" (Eshelman, 2016). It offers a central object, in this case the concept of zombie coding, and then builds a frame around it, (in this instance the artist statement and video), which is rigged to make the truth of the work irrefutable. Specifically, this statement creates a frame in which zombie coding transcends its rather prosaic nature and becomes the vital subject of a philosophical enquiry that sheds light on some of the ways capitalism shapes creative production. This is "a performative tautology that allows endless circulation of cognitively dubious, but formally irrefutable metaphysical figures within its boundaries" (Eshelman, 2016).

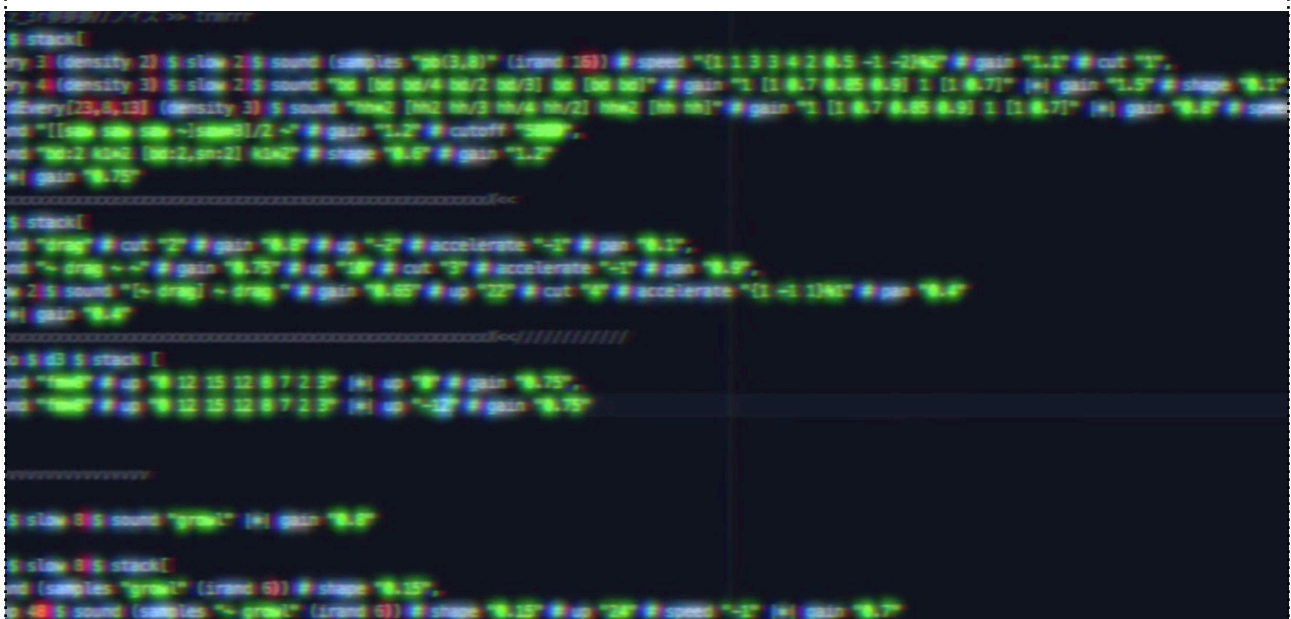
The focus of the statement was a critical analysis of my approach to live coding that involved mostly on pasting and then manipulating pre-written code during

3. Phase Two

performances rather than typing it from scratch. The statement argued in a deliberately provocative style that this analysis illuminated the degree to which capitalism has shaped the very foundations of my methods of music making and teased out a dialectic in which my zombie coding could be seen both as embracing the utopian, emancipatory potential of live coding, whilst simultaneously pulling it back towards formal structures in which it could be more easily subsumed by capitalism.

The final version of the statement can be found in Appendix V. This project also produced new practice-based research in the form of music made using the zombie coding technique. The nature of this technique is explained in the video it was made to soundtrack.

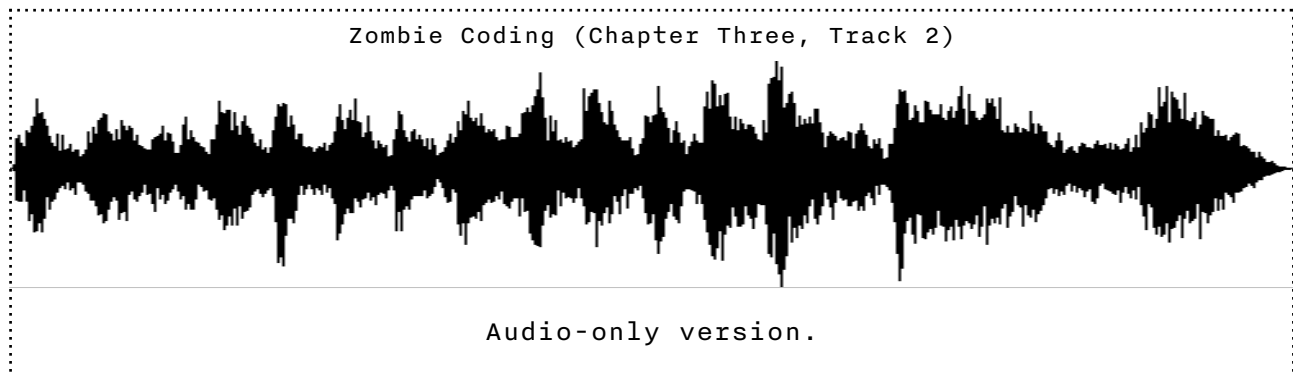
Zombie Coding



Conceptual video.

Video Title: VIDEO_17_ZombieCoding

YouTube link: <https://youtu.be/ytky6fW8bK4>



3.3.2 Further Reflections

The performatist approach was inspired by Eshelman's concept of an artwork that was able to build its own framework of meaning atop the quicksand of postmodernism's deconstructive tendencies. In the introduction to Eshelman's chapter on performatism in his book *Supplanting the Postmodern* (Rudrum, D., & Stavris, N. (Eds.), 2015), Rudrum explains this idea

...performatism recognizes that there can be no possibility of simply ignoring the postmodern assault on metaphysics... Rather, works of performatist art, literature, and philosophy are said to force us to decide in favour of these values by forcing us to decide against postmodern undecidability (Rudrum, 2015)

As a standalone piece of work, the written statement called *Zombie Coding* was not coherent in its message, certainly not in the context of an academic journal. Only with the wider contextualisation provided by its place in this study can it be useful in illustrating the development of my approach to creative production. The music made here in relation to earlier live coding work, although still being a live take, is more concise than the *Live Coding Rehearsal* piece and less experimental than the two *Tidal Test* excerpts. This comparison reflects what the zombie coding artist statement did not, which is an emerging synthesis in the music itself between algorithmic techniques and a creative practice driven by a preference for finished musical productions rather than musical

3. Phase Two

processes. What *Zombie Coding* was designed to articulate was how far this preference for finished musical productions was a symptom of capitalist realism. This project raised fresh questions for me about the nature of my work that were taken forward to the *Decomposition Theory* project covered in the next chapter.

3.4 Studio Based Music Generation

3.4.1 Intuitive Noise

Alongside the more explicitly theoretical framing around the aforementioned Phase Two projects, an effort was made to bracket off an amount of time to explore a different question. Now I had incorporated a range of original algorithmic and generative music techniques into my options for composition as well as becoming familiar with tools and concepts like TidalCycles, Max and Unity, what would happen if I let myself switch off from composing in the context of furthering this study? If I didn't compose with fully articulated research questions ahead of time, but instead returned to a more intuitive compositional approach where the final musical production is the focus, without any deeper reflection on the process?

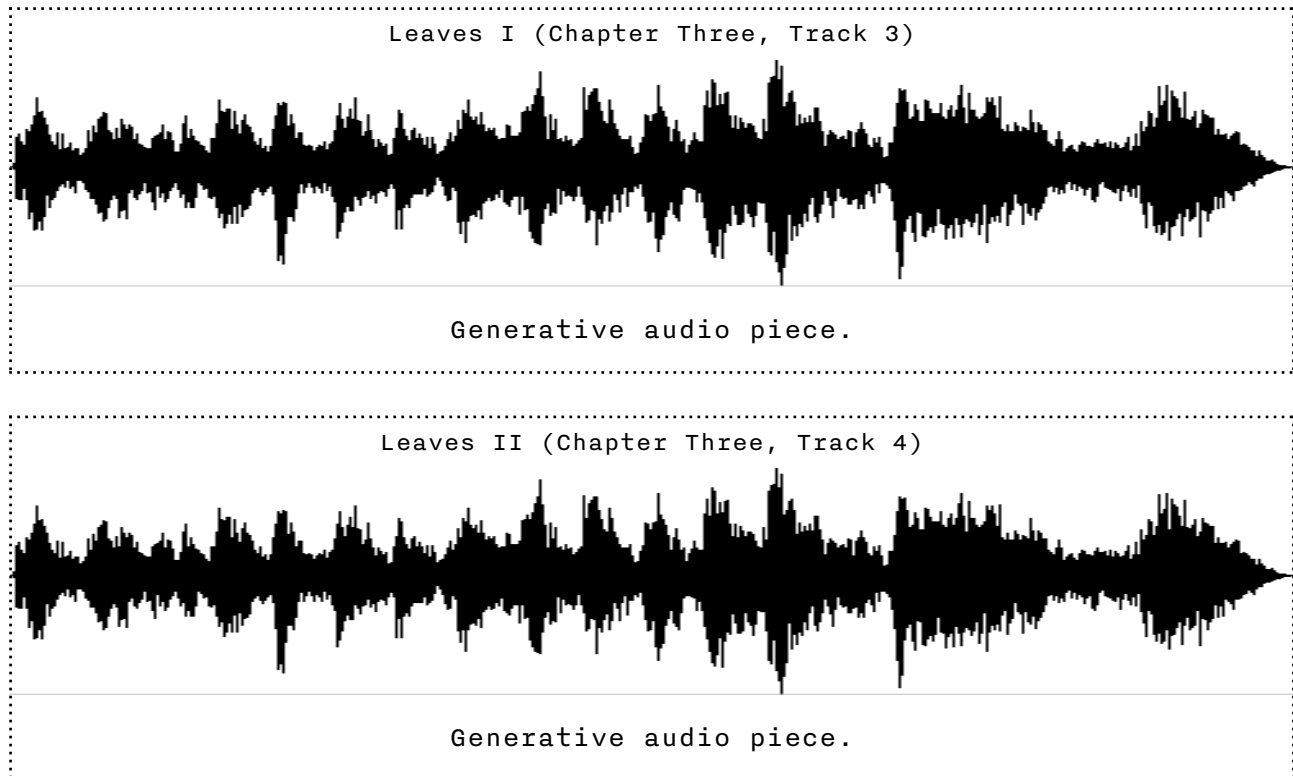
The methodology I chose to attack this question was simple: find a room with recording equipment in it, go there with my laptop and record the results.

3.4.2 Leaves

Leaves was a Max-based project built around the idea of behaviour trees, a form of nested logic often used in computer games to create complex behaviours. The direct inspiration for this piece came from the article *Behaviour Trees for AI: How They Work* (Simpson, 2014). After reading this article, instead of looking at research about how behaviour trees and similar techniques have already been applied to music (Collins, 2008), I attempted to build one directly from first principles. The autodidactic nature of this meant that instead of creating anything that could legitimately be described as a behaviour tree, I ended up with an odd generative music system in Max I called *Leaves*. Unfinished research notes are included here in Appendix VI as an autoethnographic

3. Phase Two

approach to show the thinking I had been engaging in. The research was abandoned in written form because it tied itself in knots about its purpose, struggling with the idea of intentionality in a piece of music. Whilst failing as a piece of academic writing, as practice-based research it yielded two pieces of music. *Leaves I & II*.



These two short pieces feel like some of the strongest material generated during this research. Building an unfinished, rickety music system that was to some degree a mystery to myself and then using it within a studio setting where I could achieve the level of production I wanted as the musical system remained live and malleable was a rewarding experience. Musically the system allowed me enough input to shape the mood of the songs, but surprised me with the specific nature of its output. This is music that I would feel comfortable releasing, music that I feel does not need the context of its production or knowledge of the processes through which it was made to give it a level of intentionality that justifies its existence.

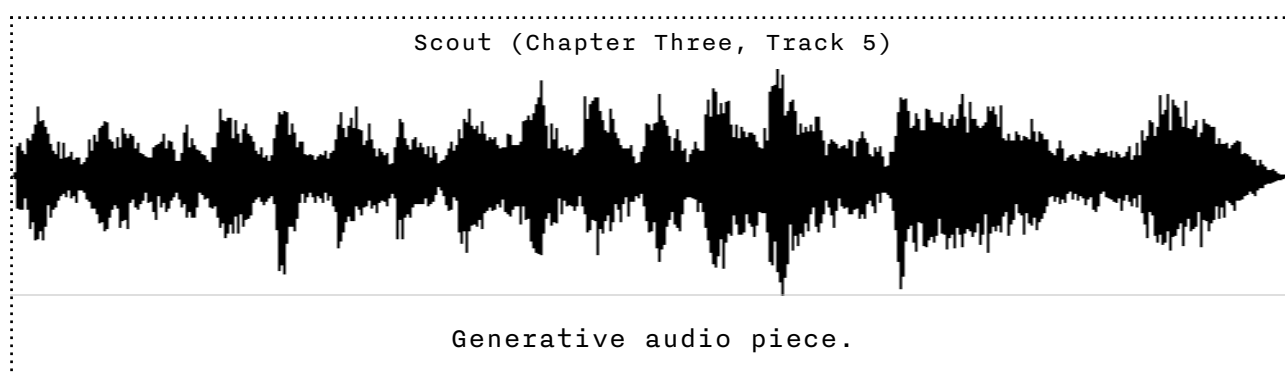
The various Leaves Max patches and javascript code are no longer available to include in this thesis. They partially exist, but became broken, fragmented and unable to

3. Phase Two

recreate the music that was recorded. This absence of documentation signposts a freedom to the mode of composition employed under this banner of 'studio-based music generation', a practice that doesn't get distracted by trying to keep track of its own process, but rather follows the music to its own conclusion.

3.4.3 Scout & GoN

Scout placed the emphasis on sound design rather than any specific approach to the temporal construction of the song. It was made by first recording a period of piano improvisation. The improvising deliberately took the form of short phrases in the same key, with gaps of silence in between. This recording was then sliced into a library of 48 different piano phrases. In a studio environment, a simple Max patch played these phrases at random, they were mixed with some field recordings I had made, and another relatively simple Max patch played a software synthesiser using some basic logic choosing from a selection of possible MIDI notes.

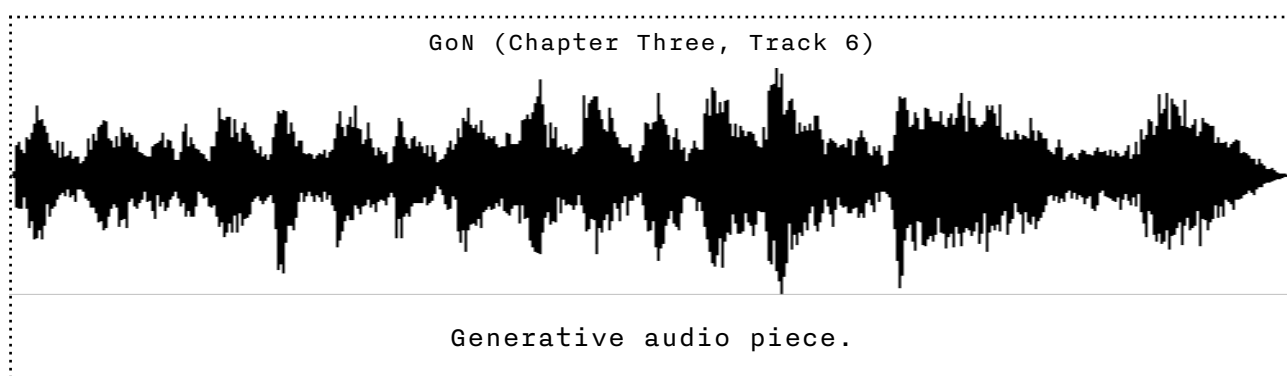


The GoN composition began using the same simple randomisation patch as *Scout*, but instead of it being filled with piano samples it was filled with wildly disparate samples, including industrial noise, choral singing, percussion, further field recordings and some of my old songs. These elements were also individually tuned and sometimes time-stretched to make sense within the context of the piece. During the compositional process, a

3. Phase Two

concrete recording of this generative output replaced the real time generation and the rest of the song was manually composed around this.

There is no strong, rational logic to say why, in the moment, I chose to build a dissonant collection of samples to create the backbone of an arrangement. It felt like a purely intuitive decision. On reflection, it shows that as generative music techniques became relatively frictionless additions to my compositional choices, they provided me with new ways in which I could begin to construct songs.

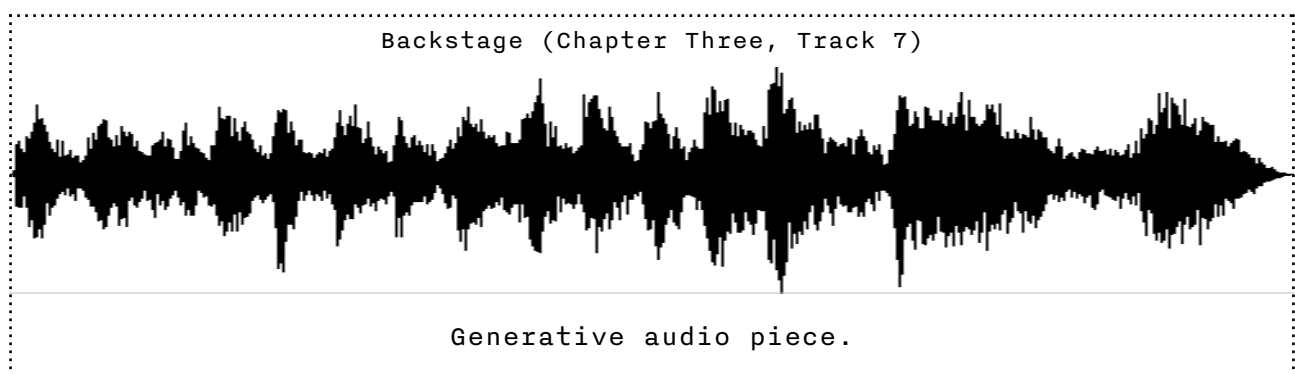


Alongside the tracks from *Leaves*, these tracks stand out to me as being of a higher quality than anything of the solo practice-based research from this study, mainly because they feel like they contain some quality, urgency and sense of intentionality that I feel a lot of the other material in this portfolio lacks. The relative simplicity of the algorithms and their non-hierarchical positioning in relation to other composition techniques really helped me reframe how I saw practice-based research in relation to how I make music. Through undertaking the previous work in which I gave myself new frameworks of understanding with regard to algorithmic approaches to music, I had absorbed these practices to a degree that they become intuitive and unforced when I am composing. This understanding reinforced the intention that this study as a whole should not be compared to other studies into algorithmic or generative music systems. As Scrivener argues, in practice-based research the work should be “culturally novel, not just novel to the creator” (2002). The algorithms used in the works concerned here are not culturally novel,

and at this point in the research not particularly novel even to me. However, the music that has been composed with them certainly is, and it is the music and this accounting of how I made it that is really the concern of this study.

3.4.4 Backstage

Backstage contains only analog-based generative music techniques. The piano track was prearranged and played via Ableton Live. A mix of this track and other elements was sent through a chain of synths, filters and outboard effects before being re-recorded. The system was designed to sit on the edge of feedback and this generated the hiss and noise essential to the character of this piece. The synth melodies came out of carefully patched feedback loops between a Moog Voyager XL and VCS3 synthesiser, and manipulation of various filter cut off frequencies, resonances and the frequencies of analog oscillators. This is a live take recorded as a stereo file; subsequent attempts at capturing the mood of this piece all failed.



This piece captures a live take that I am particularly happy with, although the nature of the system means that while the mix is not quite as clear as I would prefer, the song only exists as a stereo file and as such there is nothing that can be done to adjust the levels the way I would choose. Nevertheless, the mood of the piece captures something that I feel has worth. This is a song that I would feel comfortable releasing.

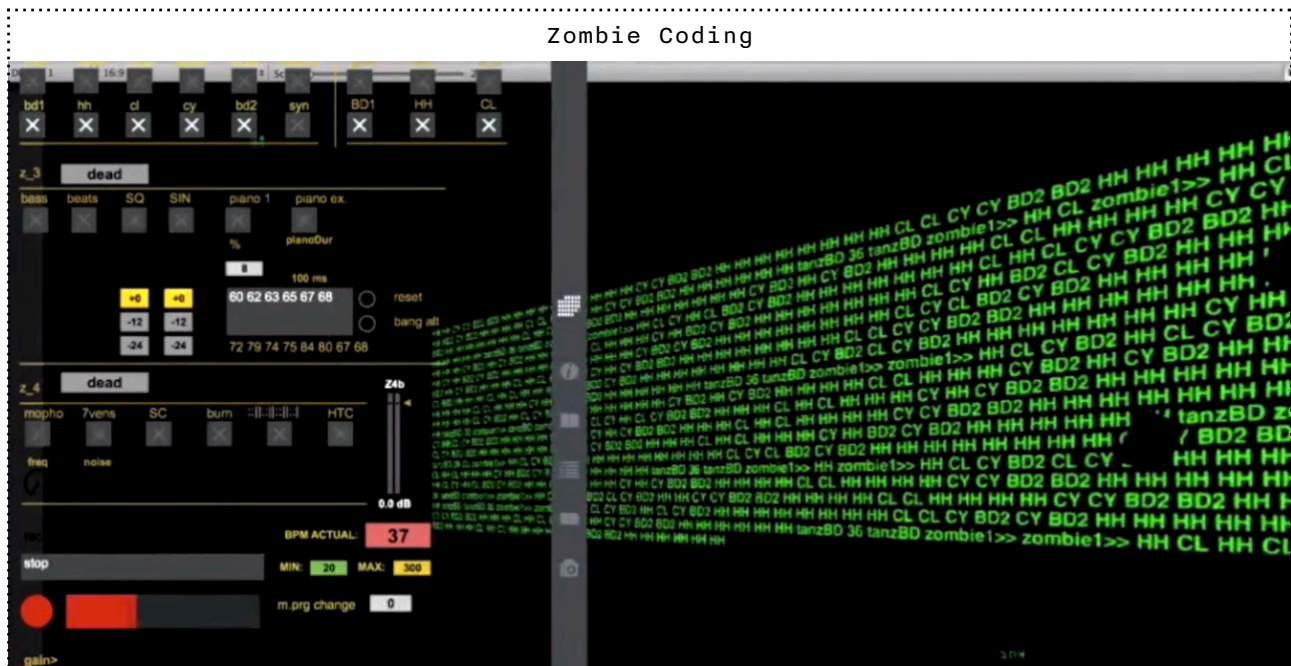
3.5 ZGEN - Generative Live Performance

3.5.1 Live Collaboration With Algorithms

The studio-based approach was a refreshing reframing of my composition priorities, signalling a return to a focus on the final musical production in my creative practice, with any algorithmic or generative music concepts in supporting roles. In May 2017 I played a solo set at an Algorave (Access Space, Sheffield, 2017) and used this opportunity to bring the same kind of reframing to my live show. This became a live framework called *ZGEN*. I knew that I would be surrounded by live coders, but was at a point where I was more interested in working with various generative techniques that were too complex to construct onstage in real time. My role in the performance had been deliberately reduced to toggling various aspects of the algorithms on and off via a Max patch. Because toggling switches would not produce much of a spectacle for the audience, I built a visual system in Unity that took the data from the decisions the algorithms were making and represented that data as if it was code being input live, even though I wasn't actually typing at all. The aesthetic gimmick was that although I was following the Live Coding Manifesto's demand to "show us your screens" (Toplap, 2010), my visual system made it appear that my computer was in the constant process of crashing whilst seemingly automating the music under its own steam, my own mouse clicks on toggles lost inside the glitchiness of everything else. This turned out to be ironically prophetic as during the performance itself the computer suffered from unexpected technical problems that led to disruptive audio stutter. These can be heard on the recording and although it remains painfully obvious to me, given the combination of the visual style, volume and nature of music I was making in the first place, it all additively fed into an accurate manifestation of

3. Phase Two

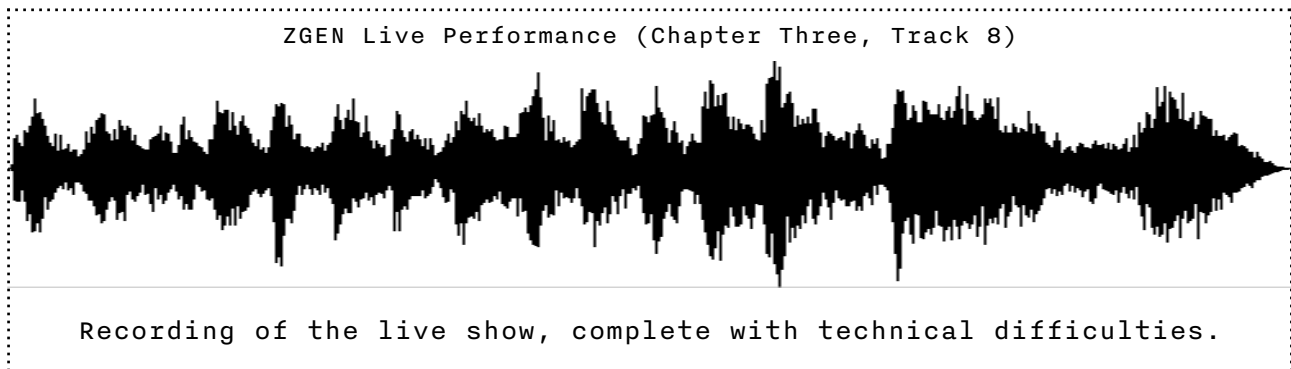
a glitch aesthetic (Hainge, 2013) and it was not clear if anybody in the audience actually noticed the music was broken.



ZGEN VISUALS AND DRUMS TEST

Video Title: VIDEO_18_ZGEN Visuals and Drum Test

YouTube link: https://youtu.be/riXypD_k8Ms



Technical problems aside, I felt like some of this music had potential but overall was not fully formed. Unlike the other research in this chapter, away from the safety of the studio environment, an amount of pragmatism was involved to make sure that the A/V would work as a show as a whole. Even with current technology running generative visuals and audio from a single laptop can be unreliable. Choices were made in terms of production (for example, all the songs shared the same set of synthesisers, reverb busses and a

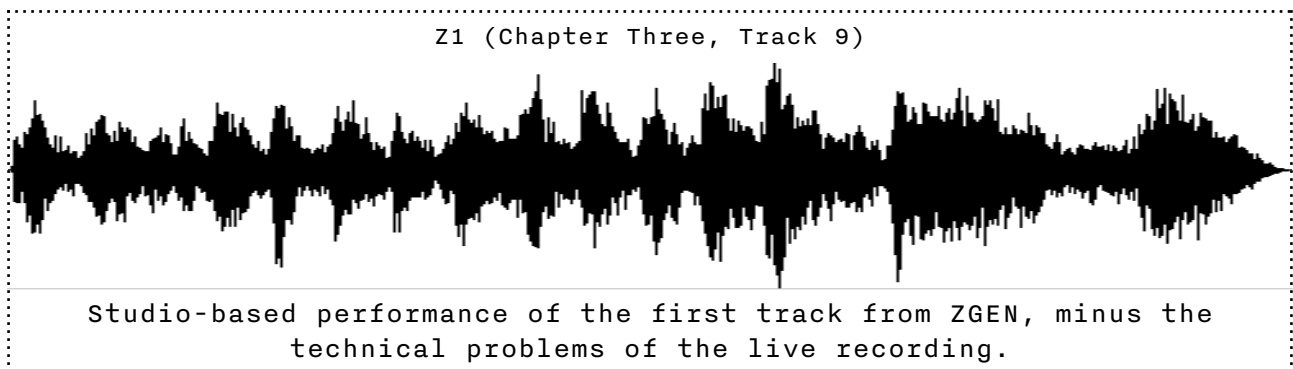
3. Phase Two

drum machine) that wouldn't have been made if these were intended to exist as standalone songs. While ZGEN worked as a one-off live set (glitches and all), it was not intended for repeated listening. In conclusion, it was a limited success and some algorithms were carried forward to the much more ambitious *Decomposition Theory* project covered in the next chapter.

3.5.2 Z1

A brief examination of the first track from the *ZGEN* performance shows how I was approaching composition at this point. The song runs inside a Max patch which drives several patterns that in turn generate MIDI pointing at a hardware drum machine and mono synth as well as three software synths and two software samplers. One software sampler is muted, triggering only a silent impulse which is side-chained to a compressor. The entire mix of hardware and software elements is sent to a chain of FX pedals culminating in a long reverb. This chain is then passed back into the mix through the side-chained compressor.

Unlike earlier experiments such as *Full Bleed A/V* where the live show didn't leave much space for performance, this setup featured miscellaneous hardware that could be manipulated in real time. It allowed for a collaboration in which the algorithm took care of driving the song and my performance could be concerned with curating the sound design of the piece in real time.



3. Phase Two

This song has an interesting character and worked well at a loud volume, especially in a gig environment. It does not have a huge amount of subtlety or variation. I could imagine a shortened version existing in a releasable form but as it stands, it is not something that I feel is particularly strong.

3.6 Chapter Summary

The projects in Phase Two showed that setting out a dialectical framework to employing algorithms to make or present music would be a valuable next step in this study. *Fully Automated Luxury Composition* showed that algorithms could open up possibilities that human composition could close back down and shape into songs. The accompanying essay (available in Appendix IV) provided the theoretical foundation for this turn to dialectics, using Jameson's notions of utopia and the utopian impulse as an entry point.

In contrast, in *ZGEN* and the songs that emerged in the studio-based experiments, the algorithms were driving the song structures, while as composer-performer I strived to open up spectral possibilities within them. As collaborators, neither composer or algorithm fully committed to their roles, and within the same multiphase compositional process this push and pull might be reversed.

As discussed in the previous chapter, the research conducted in Phase Two was happening more or less in parallel with the latter period of the *No Man's Sky* project where 65daysofstatic were working with generative musical systems designed by myself and the band, and these systems were beginning to expand beyond the parameters the game required us to work within. Taken together, the work in Phase Two, with its theoretical frameworks and emerging dialectical approach to composition converged with the ambitious and heuristic practicality of the music and musical systems emerging from *No Man's Sky*.

The next chapter, *Decomposition Theory*, combines these two strands of research into a methodology and tests it in a context fully immersed in the domain of popular music composition.

4. Decomposition Theory

4.1 Composing Decomposition

4.1.1 Ideas/Manifesto

Decomposition Theory (DT) is a term I coined to describe a methodology of working practices by 65daysofstatic, who were seeking to create music as a band whilst simultaneously attempting to subvert, expand or antagonise the forms usually associated with bands and the popular contexts in which they tend to be found. The debut of this methodology was a series of live performances in Sheffield (65daysofstatic, 2017) by the same name. The *DT* manifesto acknowledges that the entire concept of *Decomposition Theory* is a construct built around a very specific artistic practice, which is ‘being 65daysofstatic’. In this respect, *DT* is not a directly shareable technique and inspiration for it can be found in Jameson’s idea of “postmodern artistic singularity effects” (Jameson, 2003) covered in Phase Two. The manifesto, incomplete, was written only for ourselves as part of the development process.

Decomposition Theory Manifesto

1. Algorithms cannot invent the future. They can only mimic the past. But faster.
2. Algorithmic, generative music tends toward infinity. Yet the finite experience of the listener negates any possibility of a truly infinite music. It can only ever be experienced in fragments. Musical fragments: also known as songs.
3. The popular conception of songs, of albums, bands, performances: these are compromised forms. All shaped by the pressures of capitalism.
4. Yet music thrives in these contexts and the communities they create. Music unshaped by its relation to capital is increasingly hard to imagine, because we exist in relation to capital too.
5. To decompose anything, you first need to create it.
6. Frameworks for Decomposition Theory should exist in a space between states. Build them as open, endless landscapes of algorithmic processes; confine them with notions of popular music.

4. Decomposition Theory

7. When performed, Decomposition Theory is a subtractive process. It is a human unmaking of the algorithmic infinite. A curation of unpredictable processes. Fragmentation as song sculpting. Just enough pulled from much too much. Slices of time.
8. Decomposition Theory applies to all associated forms. This is not about redefining what kind of band we are. We are saying being in a band should be a continuous process of unmaking what a band is.

The three following points were not included at the time. Any ideas of formalising a manifesto for public release were abandoned, but these still serve to articulate some of the influences and goals of this research practice:

- Only the dismantling of capitalism can truly bring about a new conception of music.
- The aim of Decomposition Theory is to map a geometry of noise. To reveal the edges of the box that confines our imagination as to what musical form, bands and performance can be. Illuminating the box against itself to see how it traps the light inside.
- Decomposition Theory is not improvisation, nor is it reliable in its form. It is a dialectical process of production and destruction. A way to use atemporality against itself by wielding our own history of creative production in the form of weaponised algorithms to create new spaces of possibility, and filling them with noise.

Early discussions were based on *DT* being a fully automated music installation, but through conversations about the nature of algorithmic composition between members of the band it evolved into an installation-performance, a wary *détente* between algorithms and band members with the latter innately distrustful of the former.

4.1.2 Development Process

Inspired by the form of *ZGEN* (see section 3.5) we had initially envisaged an algorithmic superstructure that would run the arrangement and flow of a whole live performance while the band would focus primarily on sound design and curation of output. Several aborted

4. Decomposition Theory

approaches were taken in the early development stages, with Max, Unity and TidalCycles all being explored as the primary drivers of the algorithmic output.

Parallel to this, more specific algorithmic tools were being built (these are explored later in the chapter). As these smaller algorithms started to be tested in the studio, we found that we naturally began to compose around them, and distinct song sketches emerged. In turn, as human-composed content for the songs fell into place, in order to keep it from falling into strict arrangements, defeating the goal of the overall experiment, bespoke algorithmic devices were designed for each song, taking as their cues a particular piece of sound design, melodic phrase, or rhythm. This method of working emerged as a conversation between human input, algorithmic output, human iteration of algorithmic intentionality and a band-orientated sound design in terms of choosing how to instrumentalise the generative MIDI streams and algorithmic audio. In this environment, the decision to send the output of a Euclidean beat maker algorithm through a chain of guitar pedals and into an amplifier was as important as the design of the code that generated the beats themselves.

Throughout this process, we were still developing an algorithm we thought might run the overall arrangement of the show. The final attempt at this was a Max For Live patch based on Markov chain logic. Each song project would be divided into an arbitrary number of states and the Markov Chain's probability would be weighted between each. As this system came online, we quickly realised that, although the different song states were a good way of building song structures that were not fixed in time or space (similar to applying the concept of chronotypology discussed in 2.4.1) triggering them with relatively basic Markov chain logic was not as effective at creating compelling flows of music as triggering the states manually, based on how we felt the emerging songs were sounding.

A collection of footage from the development process can be found in the portfolio.

4. Decomposition Theory

Decomposition Theory Development Process



A collection of footage from various stages in the development process from August - November 2017.

Video Title: VIDEO_19_DecompDevProcess

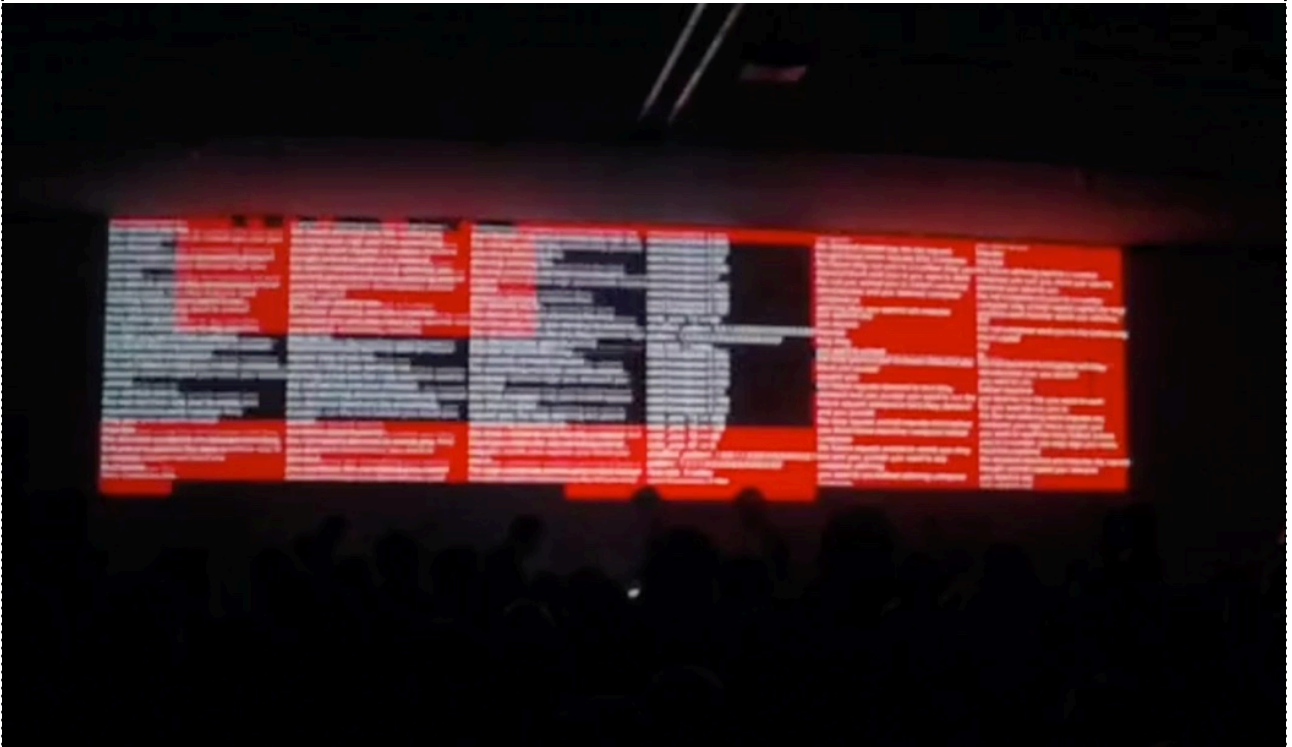
YouTube link: <https://youtu.be/7FW470EXsbQ>

4.1.3 The Practice

Audio and video from one of the three *Decomposition Theory* performances can be found in the portfolio along with a rehearsal room recording and experimental renders of audio driven by a selection of the algorithms, which serve to show the nature of how the various songs remain recognisable in different iterations despite never being generated the same way twice. In the next section a range of the *DT* algorithms are examined in more depth.

4. Decomposition Theory

Decomposition Theory Live Show



A recording of a Decomposition Theory performance from 9th November 2017.

Video Title: VIDEO_20-DecompLiveShow

YouTube link: <https://youtu.be/WLxYaPelhb4>

DecompositionTheoryLive (Chapter Four, Track 1)



Audio-only version of the live show.

Rehearsal 250918 (Chapter Four, Track 2)



Room recording of a rehearsal to test live performance of the algorithms.

4.2 Technical Breakdown

4.2.1 Euclidean Beats

Bjorklund's *A Metric for Measuring the Evenness of Timing System Rep-Rate Patterns* (Bjorklund, 2003) proposed a method for discovering "the properties of ugliness" (2003) in a given pattern and then avoiding that ugliness by distributing X number of things across a length of Y as evenly as is possible. Toussaint's paper *The Euclidean algorithm generates traditional musical rhythms* (Toussaint, 2004) applied these ideas to music and showed how almost all rhythms from music across the world and throughout history could be mapped using this technique.

A round-up of Euclidean rhythm applications from academic and electronic music communities describes a history of how enthusiastically electronic music software developers integrated these methods into experimental software (Kirn, 2011). Since then, Euclidean rhythm making tools are becoming more common in popular commercial music software like Sonic Couture's Electro-Acoustic Beat Tools (Sonic Couture, 2017) and Native Instruments' Maschine Jam sequencer module in Reaktor 6 (Native Instruments, 2017) and are often taught in music technology undergraduate courses as a way in to understanding generative techniques.

Although using Bjorklund's algorithm makes it possible to generate rhythms quickly, it is a different approach to using randomisation or chance: there is only one way of distributing 5 pulses across 16 steps using a Euclidean algorithm, no matter how many times you run it. If used alone, it is more like a rhythmical equivalent to a preset sound on a synthesiser. Furthermore it is important to note that just because all Euclidean rhythms are pleasant to listen to, it does not follow that all rhythms that are pleasant to listen to are Euclidean.

4. Decomposition Theory

The *DT* Euclidean Beat Generator is a Max For Live MIDI effect that I developed. It generates Euclidean patterns and streams them in real time in the form of both MIDI notes for the audio and OSC instructions for the visuals. The controls the Beat Generator provides to the user are:

- An ON/OFF toggle that is quantised to the bar.
- A SPARSE/NON_SPARSE toggle. Sparse patterns are generated with only a kick and a snare. Non-sparse patterns also include glitches and hi-hats.
- Manual entry to generate a pattern with chosen values for the number of pulses and the number of steps to distribute them across.
- A toggle to generate patterns automatically every X number of bars, based on an editable pool of possible values for pulses and steps.
- A mute toggle for the channel strip. This is because the generator is designed to sit directly in front of the drum samples it is driving, but in some instances it was desirable to be able to continue generating patterns that were driving other instruments while silencing the beats themselves.

When the algorithm is triggered it takes account of the arguments given to it (pulses, steps, and SPARSE/NON-SPARSE) and returns a rhythm pattern in the form of an array of ones and zeros. So, for example, a pattern of 5 steps across 16 pulses would return this:

1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0

Whilst 8 steps over 15 pulses would return this:

1 0 1 0 1 0 1 0 1 0 1 0 1 0 1

Within the main block of code that sits inside the generator, three separate collections of sounds have already been defined as kicks, snares and glitches/hi-hats. Specifically they are denoted as MIDI note numbers. For example:

```
var kicks = [36,37,41];  
var snares = [39,40,42];  
var glitch = [38,43,44,45,46,47];
```

4. Decomposition Theory

Once the algorithm has produced a pattern, it randomly picks a kick MIDI note integer (i.e - 36, 37 or 38) and a snare MIDI note integer to populate the rhythm with. It then iterates through the pattern. If it encounters a 1, it randomly places either a kick or a snare, unless it is the first beat of the pattern, in which case it will always place a kick. Whenever it places a snare, it also places an extra instruction - MIDI note '48'. This means that every time a snare is triggered, it also triggers another Max For Live device that is embedded in the drum rack on MIDI note 48. This device is a rough and ready custom-built instrument that generates a unique glitch noise with a short envelope whenever it is triggered. This is mixed at a slightly lower volume than the other snare samples and adds an extra layer of variety to every sound.

If the mode is set to SPARSE then this is all that will happen, the rhythm is complete. If it is set to NON_SPARSE, whenever it iterates through a zero, instead of staying silent, it will place a random MIDI note from the glitch array notes. Unlike the kick and snare which stay consistent within any generated pattern, a different glitch MIDI note is chosen for every zero. The samples these notes trigger are shorter, either processed hi-hat samples or else waveforms with similarly short decays. These seemingly minor aesthetic choices are as much a part of the composition as the algorithm wrangling, and the choices that are also hardest to qualitatively measure. The balance of three different kick drums, three different snares and the layering of a unique noise as well as the eight glitch samples was an artistic choice to enable creation of a sound palette that worked in the context of everything else happening in this particular song. By changing the number of possibilities in these MIDI note arrays, and changing the libraries of samples they point to, a wide range of rhythmical possibilities became available to us.

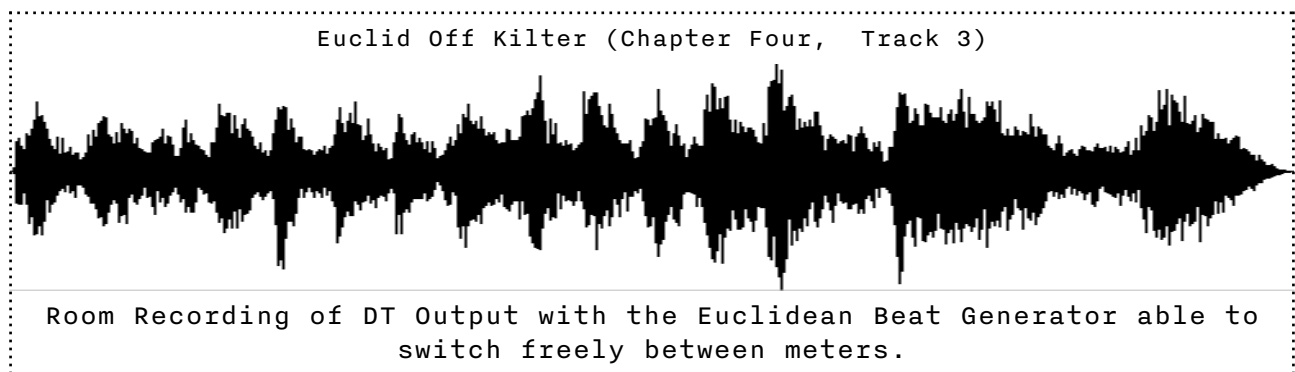
For the specific song this device was used in, which had the somewhat predictable working title of "Euclid", although the content would always be generated in real time we

4. Decomposition Theory

had decided that the broad arrangement would have two parts, each with a different tone. Both would use the same generator, but it would be playing two different kits. Therefore we used Ableton Live's inbuilt Instrument Rack, which can hold multiple instruments, and populated it with two Drum Racks. Using the chain function, switching between the kits could be done simply by triggering a clip in Live during the performance.

During early development, we considered running drum samples from a bespoke Max For Live instrument for greater flexibility but using Ableton Live's tools allowed for a much greater degree of sound design with very little downside. It was possible to route the kick drums to a separate buss, EQ every sound in isolation, add third party FX that could then be automated externally and so on. The goal here was not to make a tool that would be useful outside of this project. It did not need to be instrumentalised in a generic way or made understandable to anybody outside of the members of the band and it didn't need to work as a standalone device or even in multiple contexts. The goal was to build what was needed in order to realise our vision for this song.

For a particular type of music, this beat generator was very effective, creating immediately compelling, heavily syncopated beats when combinations that weren't based on 8 or 16 steps were introduced. In rehearsals we achieved some interesting results with this. Take, for example, this version of Euclid:



However, for the eventual live performances, these wildly varied meters within a single song were avoided. Syncing the hardware drum machine's step sequencer to beats like

4. Decomposition Theory

this whilst Ableton's clock technically remained in 4/4 was a complicated challenge and without clear reference points to know what the beats were doing or where they might lurch next, it made any meaningful live contribution by the four band members next to impossible.

The most success we had in the context of performing Euclid was to keep the number of steps at 8 or 16, and change only the number of pulses and whether the beat was SPARSE or NON_SPARSE.

A variation of the Euclidean Beat Generator also exists in a song called "Synthflood". The nature of this song was fast and focused on glitchy breakbeats, so a variation of the Euclidean Beat Generator was made that took advantage of the fact that it would always run at a rate of 16 steps per bar.

Aside from the decision to only have one choice of kick and snare, the main addition to this was a NO_GLITCHROLLS/MAYBE_GLITCHROLLS toggle. When set to NO_GLITCHROLLS the generator didn't change. When set to MAYBE_GLITCHROLLS, there was an 20% chance it would introduce a glitch roll on every step of the Euclidean pattern where a kick or snare was placed. See code below:

```
function MaybeGlitch(pos, sample, steps){

  var rolls = [4,8,12,16];
  var beatLength = 1920 / steps;
  var ran = getRandomInt(0,100);
  if (ran > 80){
    post("\naglitching");
    var rollAmount = rolls[getRandomInt(0,rolls.length-1)];
    post("\nroll amount: " + rollAmount);
    for (var i = 0; i < rollAmount; i++){

      var newPos = pos + (i * (beatLength/rollAmount));
      outlet(0,"kikSn", Math.floor(newPos), sample);

    }
    outlet(1, pos, "bang");
  }
}
```

4. Decomposition Theory

If it did introduce a roll, instead of placing a single kick or snare on that step of the pattern, it would first subdivide that step into either 4, 8, 12 or 16 sub-steps, then fill all of those steps with MIDI notes that would trigger the drum sample in quick succession.

4.2.2 Curved Beats

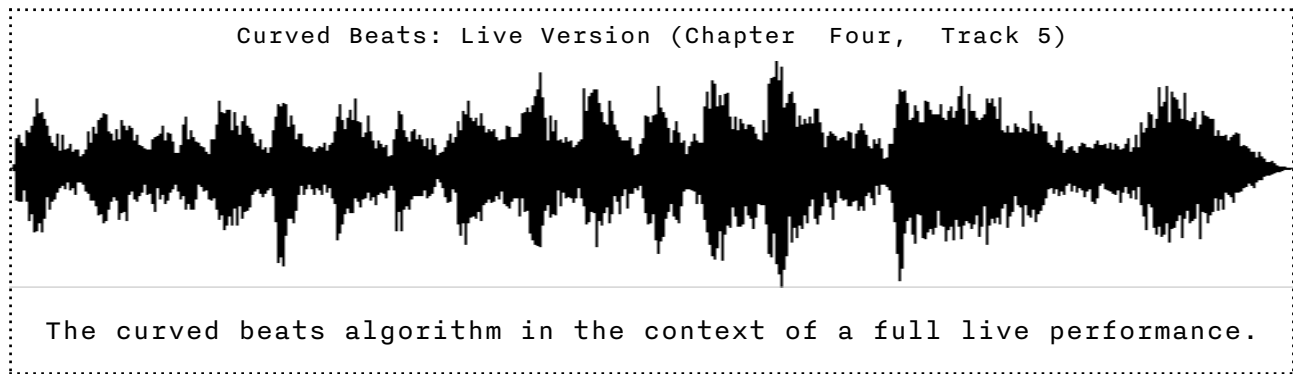
The Curved Beats algorithm was based on a recursive function that generated exponential curves and then used those curves to plot MIDI notes across a fixed length of 4 bars, producing a rhythm that began slowly and increased in speed dramatically as it reached the end of the four bars. The coefficient used changed by a small amount every time the function ran, resulting in curved rhythms of slightly different shapes. These patterns were layered up with inverse functions that began fast and slowed down to a crawl over the four bars. These two patterns of MIDI notes were then sent out to various different instruments. A skeletal piece of audio can be found in the portfolio which focuses these curves primarily on drum sounds. In the live recording, these curves not only drove the beats but were also sent to the mechanical piano and activated a gate on the live guitar, creating a stutter effect that ran in time with everything else.

Curved Beats: Skeletal Version (Chapter Four, Track 4)



The curved beats algorithm driving drums and software synthesisers.

4. Decomposition Theory



My subjective feelings on this piece of music changed over time, much as the piece of music itself has evolved and at the time of writing remains unfinished. The primary driver of the song, the contrasting exponential waves, is striking because percussive electronic music so often adheres to a rigid tempo. However, in terms of a standalone piece of music, it does not explore the nature of this kind of rhythm as comprehensively as, for example, Autechre's "Vi Scose Poise" (2001) or Aphex Twin's "Bucephalus Bouncing Ball" (1997), both of which also rely on elastic-sounding beats. As with most 65daysofstatic songs, the potential to turn this idea into something unique lies in how it combines electronic music production and live instrumentation. Later iterations of the song kept the basic overall structure, but altered the production choices, moving it away from spiky electronic and into the domain of evocative, cinematic sound design, full of tape hiss and re-amplified signal paths. The writing on this song continues.

4.2.3 Mountain

Construction of the Mountain algorithm was inspired by *Techniques for Algorithmic Composition* (Alpern, 1995), which describes Contour, a computer program made by the author as a way of generating melodies "by expanding existing melodies based on an analysis of their contour" (1995). It wasn't the specifics of how Contour generated new melodies that caught my attention, but the method of analysing existing melodies based

4. Decomposition Theory

on their shape and using those shapes to make new ones. This led me to design Mountain, which works as follows.

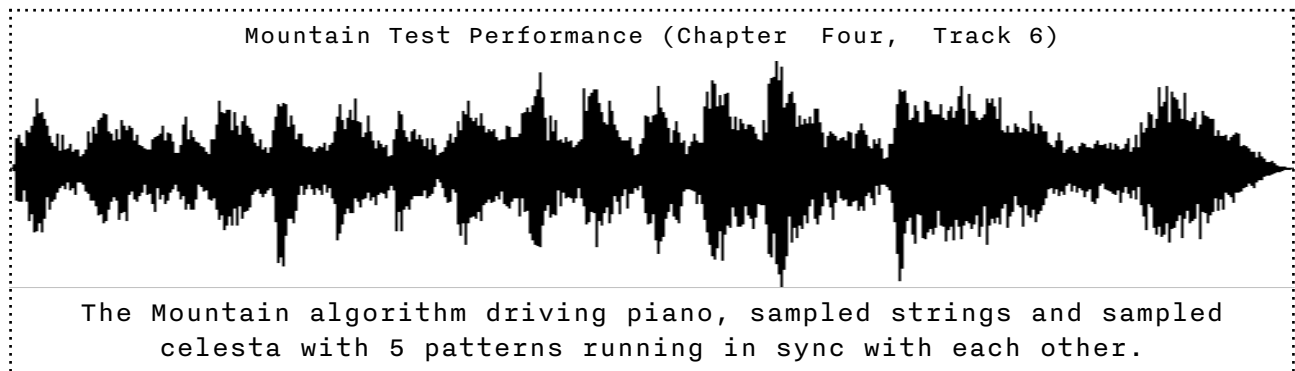
1. The user inputs existing melodies in the form of MIDI note arrays.
2. The algorithm parses each melody, calculates the jump between each note and whether that jump is ascending, descending, or a repetition.
3. From this, it generates an average jump distance and a dictionary of weighted probabilities of whether notes should ascend, descend or repeat.
4. The user specifies a root note and the total number of notes desired.
5. The initial Mountain pattern always begins with the root note. It then proceeds to build the remainder of the pattern by calculating an amount to jump based on a Gaussian distribution around the average jump distance and the weighted probabilities as to whether this should be a jump up or down.
6. If necessary it then corrects the note to the closest note in the correct key based on a standard minor scale.

This algorithm was built with the intention of feeding it existing 65daysofstatic material. A range of 65daysofstatic songs whose common aspect were the way in which they were built around long, ascending/descending chord structures were picked and their progressions were coded into Mountain.

A supplementary algorithm was made, Mountain Streams, based on the same logic except the pattern didn't necessarily have to start on the root note, and instead of creating one pattern it created four of them, each an octave higher than the previous one. Instead of feeding this algorithm with the root notes of the chord progressions from 65daysofstatic songs, the lead melodic phrases from those songs were input into it.

4. Decomposition Theory

For the *DT* performance, the patterns generated by Mountain were sent to a mechanical piano and the method of performance was based on adjusting the velocity of each pattern as well as adjusting the rates at which the patterns were cycled through. This piece of audio from the development process shows how the patterns are distributed across octaves and instruments and how iterating through them at different speeds and for different lengths can be manipulated in real time as a performance.



I feel that the piece here has a nice tone to it, although it was primarily intended as a reference point during the development process rather than something meant for a wider audience. The sampled strings don't sound particularly realistic, and the arrangement was constructed more as a test of the software than as something designed to be listened to.

4.3 Performances, Stage Setup and Roles

There were three *Decomposition Theory* performances across the 9th and 10th of November 2017. The first was an afternoon performance and the latter two shows were headline sets as part of the *Algomech Festival* (2017). The setlist for these shows was:

1. Curved Beats
2. Euclid
3. Mountain
4. Synth Flood/Z3
5. KMF

The channel list that was sent to the mixing engineer at front of house was this:

1. Laptop Beats L
2. Laptop Beats R
3. Laptop Synths L
4. Laptop Synths R
5. FX One
6. FX Two
7. FX Three
8. Piano Mic 1
9. Piano Mic 2
10. Fender
11. Orange
12. Marshall
13. Modular A
14. Modular B
15. Ampeg
16. Mopho
17. Drum Machine L
18. Drum Machine R
19. GenerativeNoise

On the stage, the instruments or areas in which the four members of 65days performed were delineated as follows:

1. Drum Kit.

4. Decomposition Theory

2. Guitar Pedals Left, Matamp Guitar Cab.
3. Control Left (Mixing Desk, Laptop, Drum Machine, Analog Synth)
4. Control Right (Laptop, Modular)
5. Guitar Pedals Right, Fender Twin.
6. Guitar, Pedals, Guitar cab.
7. Disklavier.

This section breaks down these categories to show how they were used in relation to each other.

DRUM KIT

The stripped down kit (one snare, one hi-hat), was used only on “KMF”, a song that had a small amount of generative elements in the form of a slightly modified Euclidean Beat Generator, but was otherwise a fixed song structure. Saving live drums until the end was intended to give the set a clear sense of a climax.

GUITAR PEDALS LEFT/GUITAR CAB

This was chain of guitar pedals including a sample and hold, distortion and delay effects. These pedals were fed with a send from the mixing desk on stage which both Rob Jones and myself could modify. Any combination of sounds could be sent to them, although most effective were the ‘Beats’ bus, the Mopho synth and the modular synth rack. Rather than be returned directly to the desk, the pedal chain was routed to the Marshall Amplifier and Matamp cabinet next to it. This in turn was mic’d and sent to front of house (FOH) for the final mix.

CONTROL LEFT

This area was where I spent most of my time and included a mixing desk, laptop, drum machine and analog synthesiser.

4. Decomposition Theory

The Midas Pro One desk was the hub for all the sound. The channels were mirrored to an identical desk at FOH via AES ethernet connections. The mix was grouped into busses that were mostly mirrored pre-fader, so that the band could mix the busses onstage for their own monitor mix, whilst the sound engineer at FOH could mix more accurately for the room. There were a few exceptions, for example an extra long, heavily side-chained reverb bus that was used as an instrument in its own right, and controlled by a fader, thus was sent to FOH post-fader.

The laptop in Control Left was the main controller of the whole show. This was where each song was run from a dedicated Ableton Live project. The laptop was performed by a combination of MIDI clip triggering, scene triggering and plug-in manipulation. Often, the MIDI clips were absent of any actual MIDI notes, and instead contained CC automation that would alter the behaviour or parameters of the algorithmic Max For Live patches that were running. Conceptually, this approach is somewhat similar to the Kapture Max for Live device (Bougaïeff, 2013) in that rather than triggering new melodies or rhythms, the MIDI clips were setting new states with the Ableton Live project.

Some projects would trigger the drum machine with generative MIDI notes but it was also running in sync with the laptop at all times. This allowed for beat-programming improvisation. Running in sync did not preclude standalone operation and so during the moments between the more fully-fledged songs, the drum machine could be programmed live.

The analog synthesiser was mainly controlled via MIDI, although the built-in trigger button was occasionally used to play low drone-y notes with long releases.

CONTROL RIGHT

The laptop on this side of the stage was operated by Simon Wright and ran a custom live coding Max For Live device that took typed patterns and converted them into MIDI notes.

4. Decomposition Theory

These patterns were passed to a collection of homemade and Eurorack modules, although the way the MIDI routing had been patched, these modules could also receive notes and CC instructions from the main laptop.

GUITAR PEDALS RIGHT, FENDER TWIN

The signal path of this guitar was split after the guitar pedals so that it could be fed to the Fender Twin guitar amplifier, and also into the soundcard of one of the laptops. This allowed for bespoke, often-tempo-synced effects to be applied.

En route to the Fender Twin, this processed guitar was fed via a Boss loop pedal and an Electro-Harmonix Memory Man delay pedal for further live manipulation. This allowed for unique rhythms and textures to be developed, which was especially useful for when the main laptop was loading up the next project.

GUITAR, PEDALS, GUITAR CAB

As well as being routed via a laptop, the guitar signal was also routed straight back into an Orange amplifier and Matamp Cabinet to allow for a direct, latency-free guitar sound. This output was manually controlled with a volume pedal and the guitar was played by Joe Shrewsbury.

DISKLAVIER

The Disklavier mechanical piano was tied into the MIDI interface of the main laptop. It was used sparingly and, given the unruly acoustics in the gallery, was supported by a sample-based piano through the PA to be able to cut through everything else that was happening.

4.4 Reflections

4.4.1 Decomp in Context

The project was picked up and previewed by Wired magazine (2017) and a lengthy feature interview with Future Music can be found in Appendix VII (2017). It received positive feedback from regional press (Now Then, 2017) and a positive response on social media from attendees during the performances and in the immediate aftermath, including “As amazing/weird as ever”, “Wow. That was fucking incredible”, “Great soundscape, awesome visuals!” and “Damn good set” (Twitter, 2017).

From a band perspective, it was the least enjoyable show in recent memory. 65daysofstatic have always steered clear of improvisation when it comes to live performance. As discussed at various points in this thesis, our priorities are very much looking at the show from the perspective of the audience and while we felt that it worked on that level, the situation we had created for ourselves onstage was filled with anxiety to a degree that surpassed our expectations. The combination of an awareness of the number of technical problems that could occur, the lack of control over the flow of the set, and crucially the feeling that the material we were presenting could sound so much better if we had been able to properly arrange and produce it ahead of time, as well as the feeling of being exposed standing on a stage when there was often very little to do meant that for every one of us the show was a thoroughly unenjoyable experience.

Watching back the live video, although I feel there are moments of interest lasting a few minutes at points when I understand that the songs are at their peak in terms of the amount or type of melodic and rhythmical elements running, for most of the set there are stretches of five to ten minutes where the music starts to feel monotonous or uncertain of itself.

4. Decomposition Theory

This initial iteration of *Decomposition Theory*, for the band, was neither a qualified success nor failure. It felt like a worthwhile experiment, but also for us it felt like we only just scraped through with a performance that was worth the audience's time.

4.4.2 The Question of Visibility

Decomposition Theory was an ambitious effort to combine the various strands of this study and tether them to the momentum of the ongoing, ever-evolving process of being part of 65daysofstatic. It was not designed only to be a standalone series of shows, it was an open-ended effort to explore the boundaries of what it could mean to be a group of musicians composing and performing together in the guise of an instrumental band.

Although hard to formalise explicitly, this emergent *DT* methodology can be glimpsed at in the relations, differences and similarities found in the audio recordings and videos collected into the *DT* portfolio as practice-based research. It points towards a modular approach to applied algorithmic composition in which bespoke algorithms are inserted into musical systems to play precise roles within a clearly defined range of parameters. Rather than attempting to build generic tools or complex superstructures tasked with composing entire pieces of music, the bias inherent in algorithm design is explicitly harnessed and exaggerated to allow the composer's identity to be enhanced through them rather than subsumed by them. This points the way to a mode of composition in which the application of algorithms might be visible only to the composer and not possible to detect by the listener, in contrast to the current convention at algorithmically-centric live shows like Algoraves. This opens up several areas of further study. Firstly, there are direct ways in which *DT* can be evolved within the same template by creating more algorithms and applying them in different human-curated musical systems. Secondly it raises the question of how visible the application of *DT* as a

4. Decomposition Theory

methodology needs to be from the perspective of the audience. For Algomech, a festival of algorithmic and mechanical art, the process was made as transparent as possible in press releases (Algomech, 2017), communications from the band on their website and social media (Facebook - 65propaganda, 2017) and in the mechanics of our visual system which was tied directly into the generative output and designed to show the patterns and data in real time both as literal, text-based data and graphical representations of the patterns. This all made sense thematically, but the question remained. The work in the following chapter explores to what degree possibilities might be opened up if the methods are either kept hidden or no extra effort is expended in making them transparent.

4.5 Chapter Summary

Decomposition Theory was an ambitious effort to combine the various strands of this study and tether them to the momentum of the ongoing, ever-evolving process of being part of 65daysofstatic. It was an open-ended effort to explore the boundaries of what it could mean to be a group of musicians composing and performing together in the guise of an instrumental band, an aim to formalise or conceptualise a practice that should be understood as something broader in scope than this specific series of shows.

This distinction is important to be able to analyse the *DT* live performances on their own terms, and to understand *DT* as a methodology that is not specifically tied to creating live shows in this manner, or even necessarily tied to algorithmic music at all. As the manifesto declared, for 65daysofstatic *DT* as a practice was not just about redefining what kind of band we are, but a conscious, continuous process of unmaking what a band is. A dialectical process of production and destruction.

DT as a methodology is looked at further in the following chapter, which includes not only evolved versions of the *DT* performances, but other, quite different output from myself and 65daysofstatic. This other material shows how the theoretical and technical threads of all the phases of research throughout this study can be brought together under the frame of *Decomposition Theory*.

At the beginning of this thesis, the central research question proposed was ‘how can new forms of expression and alternative modes of composition be utilised by bands and solo artists working in the commercial popular music industry?’ The research of previous chapters was made to answer specific subsets of this question. In contrast, the *DT* performances attempted to bring together everything learned so far and answer this central question head on.

4. Decomposition Theory

The performances themselves can be appraised from different perspectives. From the band's point of view, in terms of feeling comfortable onstage, they were a failure. The lack of agency and uncertainty about where the songs were heading from moment to moment created a fear that the performance might lapse into self indulgence. It also generated a frustration born from the feeling that as a band, we would be able to make music with these systems more effectively if we didn't have to do it in realtime on stage.

Despite these drawbacks, the performances were successful in other ways. Firstly, they showed that there is space in the context of a working band performing a headline live set to explore radically different approaches to what is generally expected in this environment, while still creating a recognisable, accessible experience for the audience.

Secondly, they developed an iterative approach to composition in which 65daysofstatic initially developed musical systems based on algorithmic and generative music techniques and then went on to 'decompose' within them by curating and sculpting the algorithmic output.

Thirdly, all the algorithms that were running live during the performance were linked directly to the visual system. In section 2.4.2, the experiences of working on *No Man's Sky* had raised the question of whether a manually composed, finite illusion of 'infinite music' might have better represented the idea of infinity than the actual, mathematical infinity of the generative in-game soundscapes. Somewhat analogous to this, The *DT* visuals took this question onboard and looked to balance a true description of the musical processes at play with more aesthetically pleasing or poetic interpretations of the various algorithms that were running at any given moment.

Through these performances, the emergent *DT* methodology can be glimpsed at in the relations, differences and similarities found in the audio recordings and videos collected in the portfolio as practice-based research. A modular approach to applied algorithmic composition was undertaken in which bespoke algorithms were inserted into

4. Decomposition Theory

musical systems to play precise roles within a clearly defined range of parameters. Rather than attempting to build generic tools or complex superstructures tasked with composing entire pieces of music, the bias inherent in algorithm design was explicitly harnessed and exaggerated to allow 65daysofstatic's identity to be enhanced through them rather than subsumed by them.

This points the way to a mode of composition in which the application of algorithms might be visible only to the composer and not possible to detect by the listener, in contrast to the current convention at algorithmically-centric live shows like Algoraves. This opens up several areas of further study. Firstly, there are direct ways in which *DT* can be evolved within the same template by creating more algorithms and applying them in different human-curated musical systems. Secondly it raises the question of how visible the application of *DT* as a methodology needs to be from the perspective of the audience. For Algomech, a festival of algorithmic and mechanical art, the process was made as transparent as possible in press releases (Algomech, 2017), communications from the band on their website and social media (Facebook - 65propaganda, 2017) and in the mechanics of our visual system, tied directly into the generative output and designed to show the patterns and data in real time both as literal, text-based data and graphical representations of the patterns. This all made sense thematically, but the question remained - to what degree might new possibilities be opened up if the methods are either kept hidden, or no extra effort is expended in making them transparent? The work in the following chapter explores answers to this question.

5. Post Phase

5.1 Introduction to Post Phase

When developing *Decomposition Theory* with 65daysofstatic, it had been my intention for it to be the final piece of practise included in this portfolio. It served to tie issues raised here about algorithmic techniques and new forms of expression for bands into the concerns of popular music live performance and composition, whilst the theoretical ideas about atemporality, postmodernity and how the mechanics of capitalism limit perceptions of music can be seen in the manifesto written during the development process to refine the approach we took as a band.

Nevertheless work undertaken since this project had been completed has a distinctly different quality to it, as research-based intentions retreated and intuition returned to the fore. As such, it demonstrates the development of *Decomposition Theory* as a coherent and evolving methodology, distinct from the specific series of performances and algorithmic techniques looked at in the previous chapter.

5.2 Decomp Futures

At the time of writing, eleven more *Decomposition Theory* shows have been performed by 65daysofstatic. Conversations within the band around issues covered elsewhere in this study led to a decision to build a significantly streamlined generative framework from the ground up for these performances. The notion discussed in the *No Man's Sky* chapter (section 2.4.2) that the illusion of infinity might be better than the bad infinity of open-ended generative tools influenced us to make these changes. The idea of a chronotypological approach to structure with songs built as ensembles of diachronic and synchronic events (section 2.4.1) also informed us, as did reflecting on the visibility of process in the previous *DT* performances in relation to the kind of image the band wanted to project. The significant changes can be summarised as follows:

- Rather than build a full set of algorithmic songs that will play out in a different way, we built many more sketches of songs, around 20, of which any given set was made up of around 10.
- This expansion meant that rather than building algorithms that are capable of generating radically different behaviours or sounds, they specialised in doing very particular things within different songs.
- A core collection of algorithms with relatively simple behaviour was built to be shared across sketches. For example, an algorithm that translates an incoming MIDI note into triggering the playback of a sample from a library of sounds which is manipulated based on a range of parameters. This simple algorithm (in the form of a Max For Live patch), when given different collections of samples and driven from an external source, became incredibly flexible.
- Rather than reexamine an algorithmic superstructure, or rely on a human-controlled, semi-improvised arrangement to the songs, the sketches were built upon linear timelines.

The last detail was responsible for the most fundamental shift in the musical output. In the context of popular music composition and performance, arrangement is vitally important to 65daysofstatic. The finitude of any piece of music can create a sense of purpose in its progress, a narrative arc to move through. For this run of shows, chronotypological techniques were applied spectrally and performatively within these arrangements, but not

5. Post Phase

temporally. This meant that parts of every sketch still emerged in different ways every time the algorithms ran, but the performances were diachronised by synchronic events at set times within the overall song structure. These events were pre-composed pieces of music, instructions that changed the parameters or behaviours of the active algorithms, or switched to an entirely different set of them. By removing temporality from algorithmic control, it gave the band members the ability to know where the song is headed, when. This allowed for a more involved onstage performance, manipulating the sound output in real time.

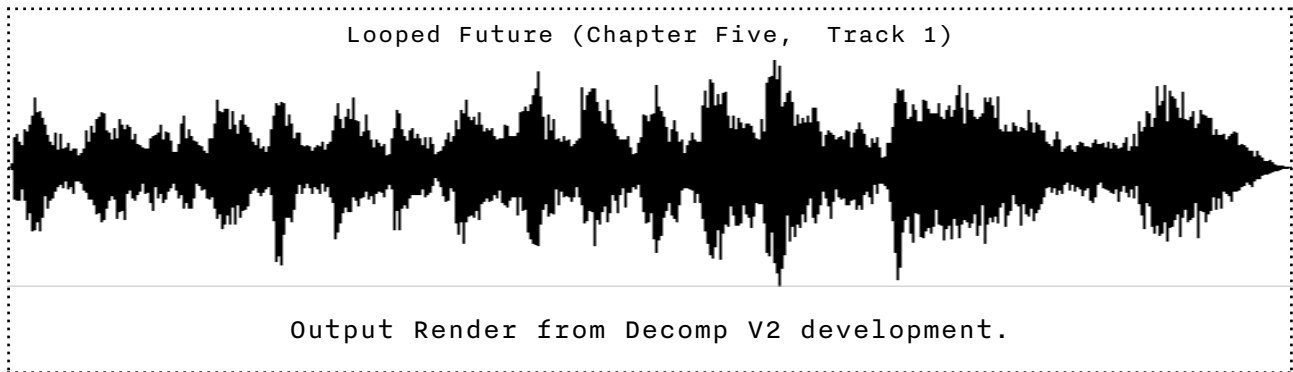
As the development process for this second iteration of *DT* progressed, logistical factors started to impact on production decisions. The debut version that took place in a gallery that gave us several days of pre-production to install the full live project that included a mechanical piano. This second version needed to run using only equipment that could fit into a number of flight cases with a collective weight of less than 80kg so we could make the European shows work within the budget available.

The biggest difference this made to our original plan was that, with less equipment to work with, to maximise the amount of live manipulation the band members could do onstage the song sketches were locked down to a larger degree than we had first planned. Whilst the FX buses, and outboard synthesisers and samplers still generated audio and sent live data to the visual system in real time, certain structural elements within each song were turned into pre-rendered audio stems. In significant ways, this made the show less algorithmically-driven than the first version, but also for us it became more focused, reliable and enjoyable to perform. Learning the song structures meant being able to respond more effectively to the algorithmic output that accompanied them.

Some experimental rendered output is included in the portfolio that illustrates the comparative focus on linear song structure to run algorithmic content within. It also shows how the output of algorithms that are more specialised in their functions bring out

5. Post Phase

specific qualities within the tracks. For *Looped Future*, algorithms are generating the beats and melodies. In *Strange Attractor*, it is the algorithms running the melodic content of the polyrhythms, but their rhythm has been built manually into the timeline.



From the band's perspective, in terms of how it worked live the new material generated during this second iteration is stronger than most of the material that emerged during the initial iteration covered in chapter four. The visuals were also rebuilt from the ground-up to better integrate with the music and run standalone (unlike in the initial version where a VJ was also needed at front-of-house to make live adjustments).

Although ultimately this run of shows was a success, there are currently no plans for further *Decomposition Theory* shows by 65daysofstatic, the primary reason being a reluctance to frame live performances around a particular technique rather than our compositions. Although it is entirely possible that the various approaches forged through the *DT* project may be included in future live shows, their visibility in terms of how the audience experiences the show would not be integral to it.

5.3 Nines

In terms of solo material written with a focus on the final creative production rather than on having a clear idea about process from the outset, and with the purpose of gathering material for a new record, a noteworthy example is “Nines”, a song written shortly after the *DT* shows and the first piece of music composed thinking that the practice-based research for this study was complete.

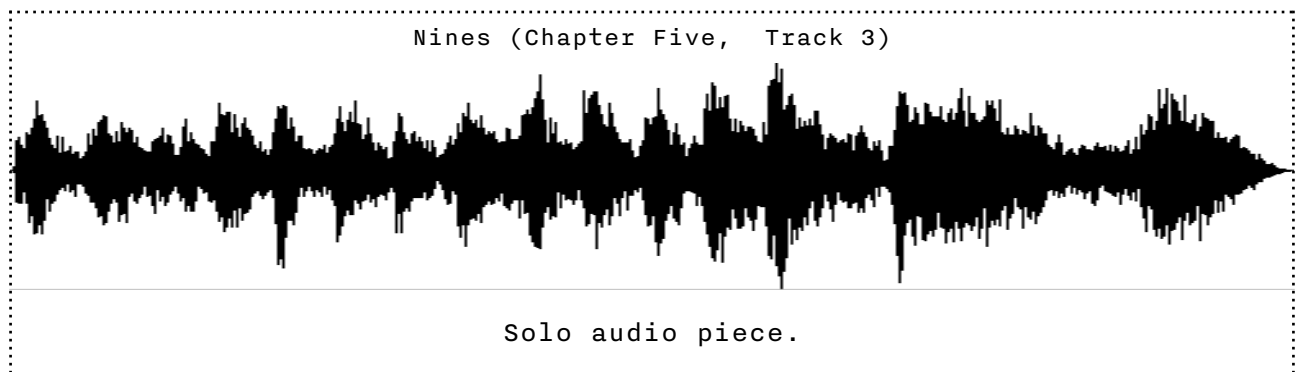
It was composed in multiple stages and although it used a generative process, the output of that process is not discernible. None of this was documented in the moment, but an approximate breakdown of the work is as follows:

1. An experimental Max patch was built that generated microtonal melodic phrases. It was quickly abandoned, but not before I recorded some of the output.
2. At some later time, this raw audio was reversed, sent through a reverb pedal and re-recorded.
3. This sound was digitally processed in Adobe Audition and saved to a folder of miscellaneous sound design.
4. Later still, I began composing the beats that would become Nines. These beats are built entirely manually with no algorithmic input or third-party ‘stutter’ or ‘beat repeat’ effects.
5. From the folder of miscellaneous sound design, the generative output was layered with the beats.
6. The rest of the song was composed in a similarly intuitive, non-algorithmic manner.

This intuitive approach to composition could describe any number of 65daysofstatic or my own songs. What has changed since before the study is having algorithmic musical techniques at my disposal. This skill has been absorbed into my work in the same way knowledge of how to program a synthesiser or how to effectively EQ a kick drum has. That is to say, it is not by itself a driving force for composition, but rather another resource amongst many others. An area of further study here would require the documentation of a wider range of songs to see when in the process of building a song algorithmic techniques are applied. Do they lend themselves only to the early stages when a composer is looking for an “Identity Idea” (Pohjannoro, 2016) to build a song upon? Or

5. Post Phase

might there be opportunities for algorithms to play a role in structuring the final song form?

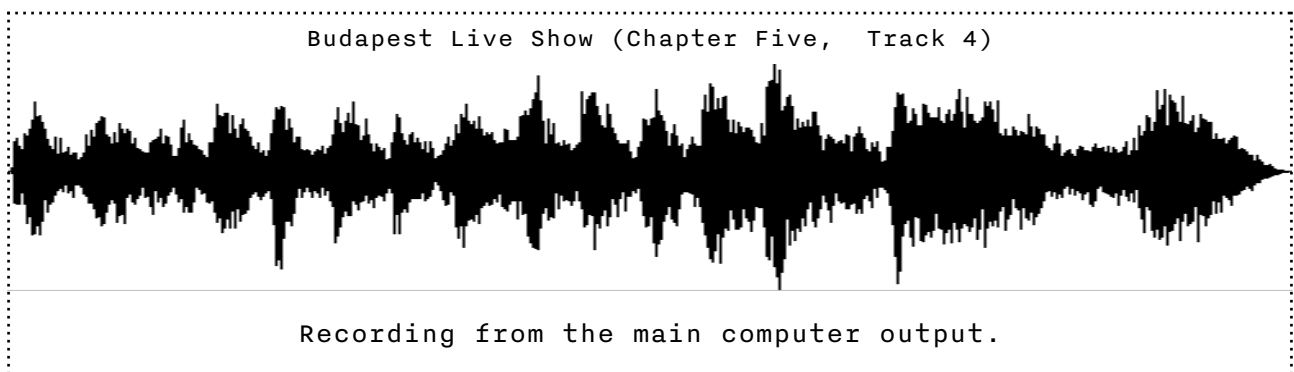


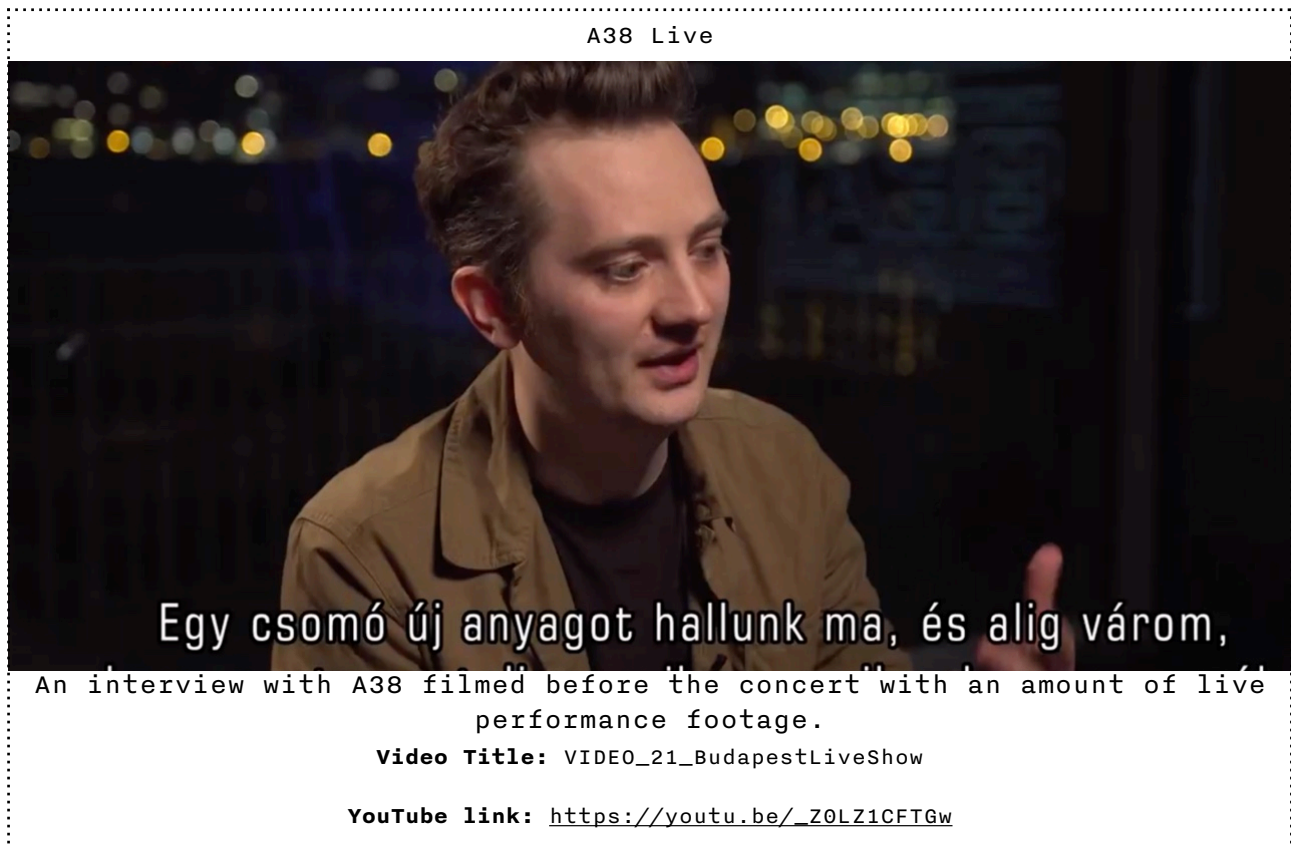
Writing *Nines* over the course of a cold evening in late 2018, at a moment when I thought the practical work for the thesis was behind me, was both a release and a reward. After long months battling with algorithmic approaches to automating the creation glitchy beats and unusual rhythms for the *Decomposition Theory* project combined with a conscious, practice-based research approach to composing, the return to an heuristic mode of composition felt like being freed from limitations I had put on myself. All the beats in *Nines* are programmed, sliced and processed by hand, a craft that I had been practising for decades and an approach that instantly made me feel closer to the music I was making than most of the other projects here. Although the sound design mentioned in points 1-3 began with some algorithmic experiments, once that audio existed, the final sound was built up from individually processed elements rather than sculpted or curated from a library of generated material. The subjective feeling experienced during this composition can best be described as a feeling of reclaiming a more fundamental intentionality. An intentionality that could be articulated only through the finished piece of music, and something that felt closer to accurately expressing my thoughts on music than the bulk of the practice-based research that preceded it.

5.4 Budapest Live Show

In January 2018 I played a solo show in Budapest at the A38 (Wolinski, 2018). It happened in the midst of various post-*DT* solo work, some of which was working with generative techniques, and a lot of the preparation for this show was with the goal of building it in a similar fashion to *DT* but to test out ideas that were being developed for the then-future 65daysofstatic *DT* shows in a live environment.

In the event, the more bespoke generative elements were removed from the musical side of the live show. This was partially a pragmatic decision: a last-minute opportunity presented itself to include visuals in the show. Looking at it from the perspective of the audience, watching me stand behind a laptop with visuals that were closely tied to the music seemed like a way to produce a more enjoyable show than the alternative of watching me stand behind a laptop with no visuals. Either way, there would be no clear indication that the music itself was generative or not. Given the time constraints, the decision became to either spend my remaining time overcoming the technical hurdles to include the more heavily generative musical elements in my set, or instead build a set of more linear, pre-written material and a visual system. I opted for the latter.





Reports from the audience, peers and promoters were all positive and I was satisfied with the quality of the sound, the focus of the music and the fact that I had found a method of performance that gave me freedom to deviate from the path, but also to control the flow of the set temporally and spectrally. It was simple enough to get lost in the performance and stop the music getting confused.

One feature of the live set-up was that the step sequencer of a hardware drum machine was synced to the main Ableton Live transport to allow for impromptu beat programming. Apart from occasionally adding hi-hat patterns to already existing beats, this largely was left unused and the 16 LEDs stepping through the patterns in time to the music was most effective as an onstage prop rather than an instrument. As seen in the *Gravity* project, a noted feature of live electronic music is the absence of the audience being able to attach the physicality of the performer to the sound that is emerging. The flashing lights give a cue to the audience that tells them that this music is tied to the technology in some fundamental way, that this isn't simply the playback of an MP3. The

5. Post Phase

generative-nature of the visuals also ties into this idea, and these extra-musical decisions that needed to be taken about the nature of the performance illustrate how in the context of popular music, a complex layering of social relations and the expectations that emerge through them about what people expect from a song or live performance becomes about much more than simply the music itself.

5.5 65days.OS, KMF and A Year of Wreckage

In April 2019 65daysofstatic launched a year-long subscription via the Bandcamp website. This project, titled *A Year of Wreckage* (65daysofstatic, 2019), is based around 12 monthly releases of exclusive material. The material is a combination of old and unused songs, various algorithmically generated outputs from the *DT* project and new productions. The project is framed as a way of trying something different to the conventional approach of bands releasing a full length album every two to three years along with a few singles and perhaps a follow-up E.P, and then going on tour to support it. *A Year of Wreckage*, as well as working as a collection of releases in its own right, is designed to act as context for the latest 65daysofstatic album *Replicr, 2019* (2019) and provide an insight into its development process. From the band's website:

We have written a new album. It's great, like sad rain in a nameless neon city, like operating systems becoming sentient in their death throes, like drinking jet lag or flowers pushing through concrete. That will happen in the autumn, and please do feel free to wait for that in a state of extreme anticipation.

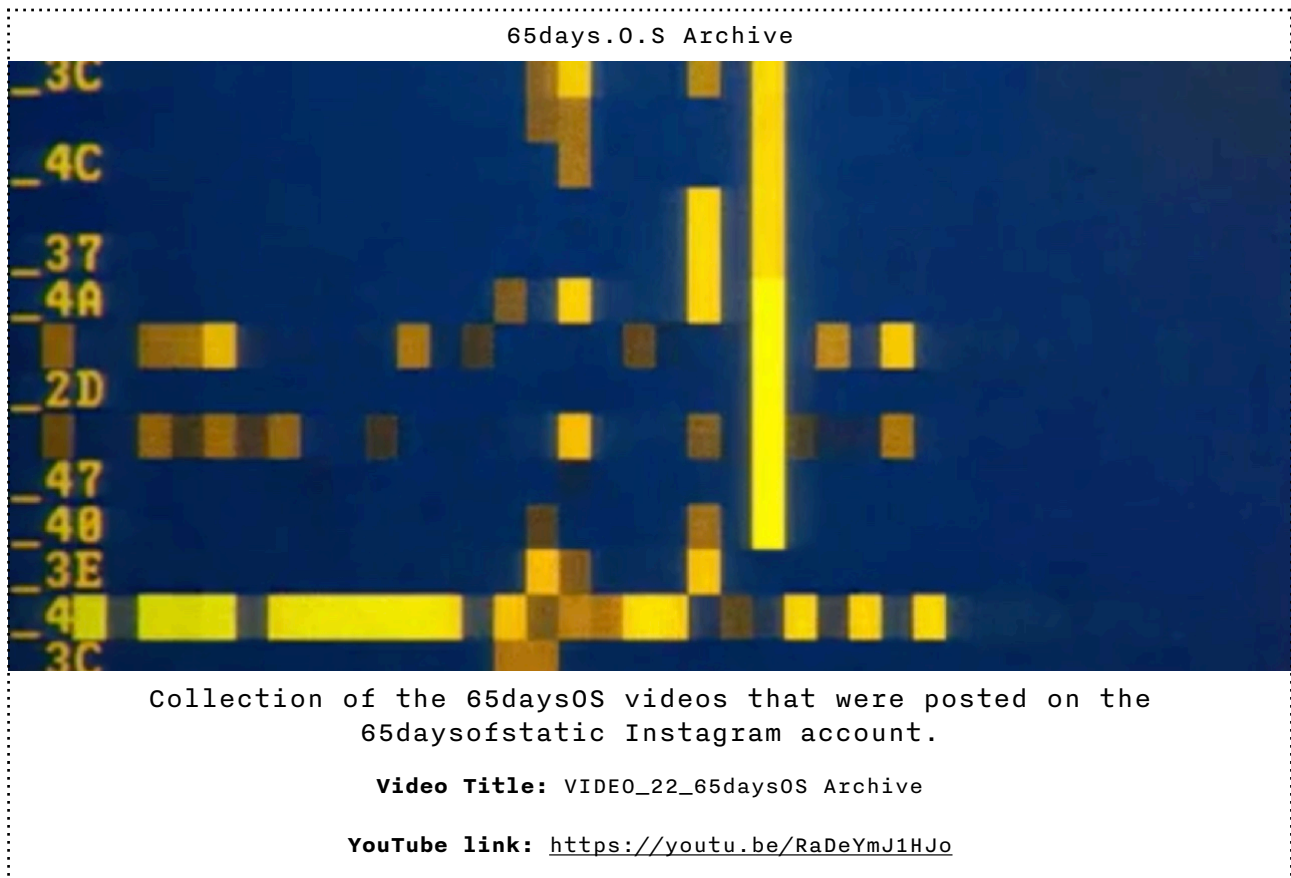
The album wouldn't exist without all that backstory, and for once that's worth talking about, or at least we think so. For that reason we're launching this subscription. This is where you can see behind the scenes of the last three years of noise and experimentation as well as the ongoing work we're still doing.

The subscription will get you an entire new library of 65music, consisting of loosely themed monthly releases curated from a recent and still growing archive of primordial 65sound palettes, a catalogue of (computer) error and guitar glitches, drone ambience and unexpected breakcore math insanity, bleak strategies and escape tunnels. (65daysofstatic, 2019)

In the lead up to the announcement of *A Year of Wreckage*, a series of videos were made for Instagram. Rather than editing music or video to fit the formats imposed by Instagram, I made the decision to lean into the form and create bespoke audio-visual projects tailored to fit the platform. This meant accepting that content on Instagram was somewhat ephemeral, and so the quantity and regularity of videos needed to be valued

5. Post Phase

as much the quality of the videos themselves. Furthermore, Instagram limited videos to a maximum of one minute at a maximum resolution of 1080 x 1080 pixels.



Using Unity and an open source package called PhiOS that provides an ASCII aesthetic within the Unity framework, I designed an audio-visual system that evoked an old computer operating system. Using only bitmap text characters and the ASCII effects built into PhiOS, the intention was to make an environment that would allow me to produce a lot of videos quickly and continuously to a high enough standard to release on the 65daysofstatic Instagram account. The system was based on three distinct processes that ran in sync with each other: audio, MIDI and screen elements.

In early experiments the audio was entirely generative, using a Unity asset called Audio Helm (2019) to allow tempo-clocked synths and samplers to run inside of Unity. However, with the ultimate goal of creating videos rather than consciously trying to answer a research question, it proved more efficient to write the music in Ableton Live to

5. Post Phase

a much higher standard than would have been possible within Unity, and then importing the finished audio as a pre-rendered wav.

In Ableton Live, alongside writing the audio, a MIDI track was also created. This track was sometimes a duplicate of a MIDI track from a software instrument, sometimes it was a duplicate with extra MIDI written into it, and sometimes it was created from scratch. The role of this MIDI track was to run in sync with the audio inside Unity. The MIDI notes in this track would then be broadcast to the screen elements in time with the music. The screen elements were a set of scripts and patterns of various complexities. These screen elements shared common properties that allowed them to listen to specific MIDI notes, or a range of them, and interpret these notes as triggers or values to adjust how they appeared on screen.

Building the project in this way meant that more screen elements could be added as the project went on and therefore the videos evolved in complexity over time. The first release from *A Year of Wreckage* was the *Kazimir* E.P (2019) and the lead track “KMF” (2019) was released with a music video that scaled up the 1080x1080 65daysOS videos of Instagram to a dense, 4K version for YouTube. To scale up the content as well as the size of the video, more screen elements were made and MIDI notes were used not just as triggers for the elements to visualise them in one way or another, but also as control signals to switch various elements on or off, allowing for a controlled variation to better fit the dynamic of the song.

5. Post Phase



A fair criticism could be made that it is disingenuous to create this text-based style and visual framework to give the illusion that it was the '65days.OS' (as in 'operating system') that was algorithmically generating the audio and the visuals as a live and dynamic audio-visual system. Combined with 65daysofstatic's association with the live coding community, and surface-level similarities to the general aesthetic of live coding videos and visuals (see Alex McLean's regular YouTube live streams (2006) or Renick Bell's text-based live visuals (2013), it is not unreasonable that a non-specialised audience might believe that 65daysofstatic had built not just their own musical algorithms, but an entire operating system to run them inside, when in fact the visuals are entirely reactive to manually-programmed MIDI tracks that are running simultaneously alongside the audio.

Although this challenge never surfaced from anybody in the live coding community, it remains true that the output is reminiscent of research from the algorithmic music literature on code visualisation, such as Charlie Roberts' *Real time Annotations &*

5. Post Phase

Visualizations in Live Coding Performance (Roberts, 2018), even though from a technical standpoint, the work within *65daysOS* is all smoke and mirrors. It does not contribute to knowledge in this respect. Its value, like the rest of the work in this Post Phase chapter, is in contrasting it with research earlier in the thesis and showing how much the role of a band is tied up with social relations and how it represents itself to an audience. This contrast helps to better understand choices like the ones made in these videos, where inaccurate but aesthetically-pleasing illusions of the musical process are valued over a transparent, authentic representation of what is actually going on.

5.6 Chapter Summary

This last collection of research shows the final realisation of the *Decomposition Theory* methodology that has been developed by myself and 65daysofstatic over the course of this study.

As explained in the introduction to this chapter, the *Decomposition Theory* performances had originally been envisaged as the endpoint of this research. Given the mixed successes and failures within those shows from the perspective of 65daysofstatic operating independently of this study, it felt important to continue documenting what came next, and how the new compositional tools and musical forms this research brought to the band and my own work continued to evolve even as the projects we undertook were less consciously shaped by a practice-based research approach to composition.

The later versions of *Decomposition Theory* as a performance showed that for the band a certain amount of pragmatism won out, allowing a greater amount of control over the overall arc of the live show at the expense of the audio being uniquely-generated with every performance. Algorithmic approaches to composition were still heavily used when writing the new material that made up the bulk of these shows, but they did not run live during the performances.

Nines demonstrated how once these tools became part of a creative practice at a more intuitive level, they worked alongside more conventional modes of composition. Similarly, the Budapest show demonstrated how the new approaches established here could be applied even when logistical constraints made the more experimental aspects subtler when compared to the *Decomposition Theory* performances.

The internet-based *65days.OS* A/V pieces showed how the tools and frameworks developed throughout this study were used to design and present audio-visual works on Instagram, a social media channel that bands are more commonly seen using to share

5. Post Phase

photos or videos of live shows, documenting studio time or advertising their latest merchandise and releases, rather than as a primary outlet for their art.

A Year of Wreckage, the 12 month subscription project, demonstrated that it was possible for 65daysofstatic to make an album, *replicr, 2019*, to fulfil our desire to make a relatively short, focussed piece, whilst simultaneously having a platform for the presentation and distribution of the huge archive of material that was created throughout the *Decomposition Theory* research period and the *replicr, 2019* writing process. This material provided valuable context for the *replicr 2019* album but was still able to function as a series of releases in their own right.

Taken together, *A Year of Wreckage* and *65days.OS* show how the practice-based research approach that has been applied to 65daysofstatic's and my own work through this study has opened up new forms for creative expression. And it has done so without demanding a complete overhaul of the previously intuitive, heuristic approach to composition. At the same time, a practise-based research approach has not been able to seamlessly integrate with an uncritical, artistic one. This outcome points to a dialectical approach to composition, which is what I call the *Decomposition Theory* methodology. This is will expanded on in the conclusion.

6. Conclusion

6.1 Summary

Setting out to discover new forms of creative expression from the context of a band working in the music industry, this study documents an evolution of both practice and theoretical ideas, from the initial explorations of new technologies to the establishing of new compositional frameworks and methodologies for making music.

Phase One as a whole showed that formalising a previously heuristic approach to composition in terms appropriate to practice-based research in order to produce shareable knowledge caused immediate tensions. *Gravity* allowed for the introduction of new theoretical ideas to be incorporated into my creative production, but the music it directly led to was underwhelming. *Post Keyhole* showed that just because generative music concepts could produce compelling musical output, the extra-musical concerns of a popular music practitioner might lead to it never being used. *Full Bleed A/V* hinted at a distinction that emerged more fully in later projects between an aesthetic (as opposed to accurate) representation of compositional processes versus the actual compositional processes themselves.

The *No Man's Sky* soundtrack revealed first of all that although the technical complexities of the algorithms used were conceptually simple compared to the current state of the art, they were still sophisticated enough to generate an infinitely long, responsive video game soundtrack. It also showed the value in looking at the same project from an interdisciplinary perspective. Chapter Two provides an academic analysis of the processes undertaken for the project, while from a 65daysofstatic perspective, our approach was more holistic. As band member Joe Shrewsbury said,

We are naive in a useful way...We're not here because of our expertise in programming or coding or writing for games, we are here because we are experts

6. Conclusion

at being 65daysofstatic and making something messy and compelling and bringing that to the game. We put the emotional elements of the music over the practicalities of making it happen. We don't care what is not possible. (Shrewsbury, 2016)

Additionally the way in which the *No Man's Sky* material was translated into various styles of live performance revealed a divergence of priorities when it came to algorithmic composition as a research topic and as a tool to present music. For the Sónar performance the choice was made to present a linear, through-composed representation of the infinite possibilities of the soundscapes rather than use any generative music tools onstage. On the other hand, if a more favourable initial reception of the game had led towards more *No Man's Sky*-centric performances, the ability to generate endless amounts of *No Man's Sky* material would have put us in the noteworthy position of being able to construct a wide variety of unique performances that were derived from the output of algorithmic techniques, but actually all manually created to ensure a high overall quality.

Phase Two took onboard the possibilities and complications that emerged through both *Phase One* and *No Man's Sky*, and looked at how various algorithmic techniques might be applied to composition from a more intuitive as well as theoretically-driven practice. The results of these projects outlined the beginnings of a dialectical method in which composers simultaneously collaborated with and contradicted algorithmic composition that was in the first place driven through functions designed to further the compositional preferences of the composer. *Fully Automated Luxury Composition* looked at this from a perspective of the musical output of algorithms shaped by the composer, whilst *ZGEN* looked at the musical output of the composer being shaped by algorithms.

Chapter Four tried to pull together all the research strands whilst simultaneously contextualising them within wider issues surrounding contemporary popular music.

Based on the theory set out at the beginning of *Phase Two* it showed that, if

6. Conclusion

Decomposition Theory as a concept was limited to this specific approach to live algorithmic performance, it would bear similarities to Jameson's idea of a "postmodern artistic singularity effect" (Jameson, 2003). This is in the sense that it was the result of a unique combination of distinct realities brought together to form a one-off process. This specific *Decomposition Theory* project could have only emerged at the convergence of the last fifteen years of 65daysofstatic's output, the recent experiences on the *No Man's Sky* project, the theoretical frameworks that I had encountered during this study, the political leanings and social relations of the band and our relationships with various venues and musical event organisers in Sheffield, the band's hometown and where the performances happened. For *Decomposition Theory* to be formalised as a *methodology* rather than as a description of this particular approach to algorithmically-driven live performances, further work of a different kind would need to be produced.

The final chapter showed a return to a more intuitive mode of composition once the practice-based research approach to writing music for this study came to a close, yet also showcased new projects like *65days.OS* and *A Year of Wreckage* that utilised technical and theoretical frameworks that were developed through this research. This final work helped to disconnect the research from the specifics of the algorithmically-driven live shows, and led towards a new methodology that I call *Decomposition Theory*. A modular, iterative and dialectical approach to existing as a composer and a band, that protects the heuristic, intuitive approach it evolved from, while also creating a space for conscious, research-based experimentation and practice.

6.2 A Critical Framework for Composition

Decomposition Theory as a methodology provides a critical framework for composition, as well as new means of disseminating work. It does not necessarily need to be tied to algorithmic composition, but the pieces of research that used this approach are useful in examining the dialectical relationship that emerged from the push and pull of algorithmic and manual approaches to making music. What is shown is that, in the context of being a composer of popular music, algorithmic music strategies can be employed by looking at composition as a multiphase process through which music can be composed not just spectrally and temporally, but in terms of potentialities or multiplicities. By conceiving the act of composition as a number of phases or iterations, it allows us to add, shape and remove algorithms at different points where they can open up or close down musical possibilities. The composer can work with or against them in a dialectical relationship, giving them as many or as few responsibilities as they choose. The notion of a remix can be seen as a proto mode of dialectical composition, but compared to conventional remix practices that shuffle, rearrange or reimagine some canonical version of a song, usually confined to manipulating consolidated audio stems, algorithmic strategies can inject more granular techniques at earlier stages that precede any definitive notion of what the song already is. A composition like this can exist in a liquid state related to its own structure, ready to manifest itself in a multiplicity of forms. Because popular music is often concerned with musical production rather than abstracted compositional processes, this methodology remains grounded in the “natural language” (Adorno 1941) of popular music. That is to say, it is designed to produce music in the shape of songs. This way of composing can build stacks of two-dimensional tempo-spectral representations of a musical space. Unlike ambient generative music that offers a window onto an infinite sonic landscape through which the listener can make

6. Conclusion

their own way, these songs deliver slices of frozen possibility, each slice taking the listener by the hand and guiding them through that landscape in a compelling way. It is not only the feeling of being led through the sonic landscape that appeals to the listener of popular music, but also a familiarity toward the hand that is leading you, and its associated cultural content.

This iterative, multiphase approach to composition is at the centre of the *Decomposition Theory* methodology. It is not limited to inserting algorithmic techniques into music, but can also be seen as a way of utilising practice-based research approaches to the compositional process in addition to the heuristic, artistic instincts of the composer, without having to solve the tensions and contradictions between them.

The *Decomposition Theory* performances documented in Chapter 4 show this in action. They were conceived as a research project from the outset. 65daysofstatic wanted to find out how the techniques acquired through the *No Man's Sky* project and earlier research in this study could be applied in the context of a live band performance. This question led to the creation of the *Decomposition Theory* manifesto, the focus on process-driven composition, and a lengthy development process of generative and algorithmic musical tools before any actual composing began.

At this top level of conceiving, developing and then presenting the performances, as the project moved through the development phase of production and into questions of how it would be presented, the band began to make decisions that would not have been suitable for a project that existed purely in a research-centric environment. A notable example was our decision to scrap a particular strand of research that was based on developing a system to generate song structures in realtime onstage using Markov-chain logic. The completion of this sub-project would have been valuable in a research context to establish how far such an approach to song structure might be useful for composers. But 65daysofstatic abandoned it before it could be fully completed, measured and

6. Conclusion

documented, having sensed that the immediate results were not producing structures that were as effective as ones we were building manually. Similarly, although the musical algorithms were running live on stage and had been designed from the outset to send explanatory data to the visual system, the choice to deliberately obscure, disregard or at times even fake this data was a decision taken by the band, as we decided that the aesthetic priorities of the performances were more important than accurately representing the processes that were creating the music.

At the micro level, this dialectical relationship can be seen in the many individual elements that went in to building each track. It is visible in the balance between unknown algorithmic outputs and manually composed song elements. The moment when the practice-based research approach of presenting live algorithmic music onstage was overridden by the band's desire to give the show a sense of intentionality is also evidence of this. The effects of this particular decision can be seen more clearly in the later iterations of the live performances discussed in Chapter Five, where although algorithms were used in the building of many more song sketches, on stage they had been entirely removed from driving the songs temporally.

6.3 A New Means of Disseminating Work

Broadly speaking, the practice-based research over the course of this study shows a creative practice evolving from conceiving composition as a linear process that led to a particular goal, whether that goal was a song, album, or sound installation, to seeing composition as a modular and iterative process of building creative systems, capable of generating large amounts of music and sometimes visuals. In the case of making a particular song, an additional necessary step was the sculpting or editing down of the output of the system, in order to close down possibilities and finish a piece of music. At other times, the system remained open and flexible, keeping the work in a liquid state. Approaching composition this way enabled me to gain a deep access to my own creative process, with a new critical awareness of how I work both alone and with the rest of 65daysofstatic. This awareness gave me a granular control over all aspects of my practice at all scales, which in turn opened up new possibilities for presenting work.

An example of this is the Sónar performance of *No Man's Sky* material. Building this one-off set established that 65daysofstatic were able to produce, at short notice, show-specific, unique versions of a dozen brand new songs. This was because these songs had already been written, deconstructed, catalogued and rebuilt within various generative musical systems. They existed not only as studio recordings, but as libraries of samples, sound palettes, chord progressions and melodic phrases. Had there been a whole tour of *No Man's Sky* themed shows, 65daysofstatic would have had a choice as to how best approach them. It would have been possible to perform them with live, realtime generative music components. Alternatively, like Sónar, we could have iterated through this re-compositional process at high speed, quickly composing many shows, all with unique variations, ahead of touring them. This would have sacrificed some amount of

6. Conclusion

algorithmic authenticity from the performances themselves, but would have given the band a means of rehearsing the shows more efficiently. Additionally, this would have been a way of making sure that our creative hand was firmly guiding the progression of the show and we were not subject to the whims of the algorithm whilst on stage.

Another example of how this approach led to new means of disseminating work is the *65days.OS* system covered in Chapter Five. It was built from knowledge of algorithmic and generative music techniques, and evoked the aesthetic of numerous live coding platforms while giving the illusion that what people were seeing was a fully functioning computer operating system. The *65days.OS* software contained generative music systems based on Audio Helm and scripted logic running inside Unity, and the original plan at its inception was to build a tool that would generate music and visuals at the push of a button. As the system evolved, it became apparent that its output was much more compelling when pre-composed music was fed into it, along with MIDI triggers that would control the visuals. Although this latter path was chosen by the band for creative reasons, we plan on returning to the project and further exploring truly generative outputs in the future. Even though what began as a generative music machine had temporarily been turned back into something more like a highly customisable music visualiser, its existence meant that the process of turning short, minute-long musical pieces into fully-fledged audio-visual pieces with a coherent aesthetic became very quick to turn around. Producing daily unique videos for the band's Instagram channel became a viable proposition. This is the *Decomposition Theory* methodology in a nutshell: building a system that can generate myriad, cohesive musical (or audio-visual) potentialities, which can then be 'decomposed' back down into distinct musical productions.

6.4 Theoretical Underpinnings of Decomposition Theory

Decomposition Theory as a methodology is underpinned by several theoretical threads that were outlined at the beginning of this commentary and expanded upon throughout in respect to the relevant pieces of practice-based research. Fisher's ideas about capitalist realism provided an important impetus and helped begin formalising 65daysofstatic's desire to always search out new approaches to being in a band. As the research got underway, Fredric Jameson in particular provided a wider context for Fisher's melancholic outlook regards contemporary popular culture, and the slow cancellation of the future that we as humans (and therefore as artists, musicians), were being subjected to under capitalism. The three, unpublished addendums to the *Decomposition Theory* manifesto first seen in Chapter Four are worth repeating here:

- Only the dismantling of capitalism can truly bring about a new conception of music.
- The aim of Decomposition Theory is to map a geometry of noise. To reveal the edges of the box that confines our imagination as to what musical form, bands and performance can be. Illuminating the box against itself to see how it traps the light inside.
- Decomposition Theory is not improvisation, nor is it reliable in its form. It is a dialectical process of production and destruction. A way to use atemporality against itself by wielding our own history of creative production in the form of weaponised algorithms to create new spaces of possibility, and filling them with noise.

A succinct drawing together of the theoretical threads running through this research can be seen in these declarations. The first declaration accepts the trap of making art in postmodernity, what Jameson calls the "nostalgia mode" (1991), this endless recycling of not just musical genre, but of commodified musical forms. Despite the digital revolution

6. Conclusion

and huge potential of the internet, in popular culture the legacy forms that shape how music is consumed are still very much present. Bands are expected to put out singles and albums. They are expected to go on tour to play songs from those records. Their singles are what the radio stations will play if the band is popular enough. It is possible to point to musical artists experimenting with new forms, for example Bjork's *Biophilia* iOS app (Second Wind Ltd, 2011), or Holly Herndon using machine learning to collaborate with A.I songwriters (Hawthorne, 2019). It nevertheless tends to be the case that these experiments exist as offshoots of album campaigns. They are coopted by the record labels tasked with promoting the release of their artist, and put into service as imaginative marketing and brand building exercises as much as they exist as artistic creations in their own right.

Decomposition Theory's insistence that the dismantling of capitalism is what is required to bring about a truly new kind of music, one that does not need to be anchored by these legacy musical forms and institutions, is a militant rephrasing of Adam Harper's idea that "music cannot be free until society is". Harper goes on: "The two domains are not just intimately linked, but one and the same: music is a form of social communication. There is little point to a culture of free music-making unless it exists within a prosperous, non-precarious post-work economy" (2015). Music as a form of social communication in contemporary society is an ever evolving, shared language of mutually understood scales, instrumentation, genres, and musical structures that exists in and through capitalism. Martin Stokes calls these pockets of shared understanding "communal solidarities" (2013); for Christopher Small it all falls inside the concept of "musicking" (2011). Harper's "n-dimensional modernism" (2011) strives to create new, unthought of music, but accepts that this music needs to be grounded to some extent in the old and familiar for it be to appreciable by the listener. This attitude can be seen in the *Decomposition Theory* methodology as it seeks to push forward new forms for the creation and presentation of

6. Conclusion

new work, while acknowledging and maintaining these vital aspects of a common musical language that exists through the social relations of popular music culture.

The second addendum of the *Decomposition Theory* manifesto shows how ideas from Attali and Jameson about utopia and the utopian impulse fed into the methodology. These themes are explored in much more detail in *Fully Automated Luxury Composition*, my essay from IASPM that is outlined in Chapter Three and included in full in appendix IV. The utility of utopian thinking is not to be able to directly lead us into utopia, rather, as Jameson puts it: "Utopia can serve the negative purpose of making us more aware of our mental and ideological imprisonment (...) the best Utopias are those that fail the most comprehensively" (2005). *Decomposition Theory*, by trying to reach outside of the established forms that bands and composers are expected to work in, provides a better understanding of their limits.

The third addendum introduces the importance of the dialectical method. As Fredric Engels said, "For [dialectical philosophy], nothing is final, absolute, sacred. It reveals the transitory character of everything and in everything; nothing can endure before it except the uninterrupted process of becoming and of passing away, of endless ascendancy from the lower to the higher." (Engels, 1990). *Decomposition Theory* approaches music making in this way. By thinking about the act of composition as a multiphase process, in which a musical system is first composed and then works are 'decomposed' from it, the composer can lean into music-as-process while still producing musical objects that fit into the expectations of popular music culture. This study has shown how *Decomposition Theory* can work in terms of balancing algorithmic and manual composition. It has also demonstrated how to find a balance between academic, practice-based research approaches and instinctive, spontaneous artistic decisions.

In this methodology, individual phases of composition can produce shareable knowledge, serve to obfuscate their true nature, do both or neither. In the *Decomposition*

6. Conclusion

Theory performances for example, the research demonstrated new possibilities for live popular music, whilst simultaneously the final musical production itself favoured illusory representations of the algorithmic process over attempting to communicate any directly shareable knowledge. Explanations of the *Decomposition Theory* techniques found in this writing are detailed and transparent enough for other readers to follow and build their own algorithmic functions, but in respect to the public image of 65daysofstatic, the band spoke in more ambiguous or poetic terms. Knowledge is created, documented, and at the same time hidden or disguised.

When 65daysofstatic announced their *Decomposition Theory* shows in 2017, we described them as follows:

Writing a song feels a bit like working your way from the inside out. In the beginning, you're right in there with the noises, rhythms and melodies. There's no structure, it's not even trapped by time. It's this bright little idea that's full of potential expanding in every direction at once. But as you work outwards, you start to trap those ideas in the form of parts of a song. Songs are pretty linear structures. They are established forms that we all understand and enjoy. So a lot of the time that's great. Because making songs means making albums. And albums are great too. By the time it comes for us to make a record, we are usually outside the song. We understand what it's for and what it wants to be, and then it's sculpted into its final shape. It becomes fixed in time. Along the way, something of the original idea is always lost.

Decomposition Theory is an attempt to work from the outside back in. Or maybe bring the inside back out with us. To make what's essential to the music be the process of making it, instead of its final form. Every time we do this show, the music will change. We have made songs only as collections of sounds, logic and rules. Familiar but always different. There's no definitive versions of anything. But it's not quite improvisation either. Because that can get really self indulgent. It's more an attempt to keep all the bits we think make good songs, but without closing off the excitement of not knowing exactly what happens next.

Why are we doing this? No doubt, 65 are a long way from the frontline in any kind of struggle against capitalism and the abyss of a future it's driving us toward. But perhaps there's a supporting role in trying to imagine better futures. We are not railing against recorded music or albums or regular live shows, or what it can mean to be a band. However, these are all commodified forms. Almost all the ways anyone can relate to music these days are mediated through capital. Because we exist in relation to capital. We're all drowning in it. It's almost impossible to think outside of it. And so Decomposition Theory is some small effort to imagine a space where music is no longer shaped in commodity forms, and a live show doesn't emphasise the ritual of performance. We are un-singing our songs and un-performing on stage. (65daysofstatic, 2017)

6. Conclusion

Un-singing songs and un-performing performances. For 65daysofstatic, creating *Decomposition Theory* was an effort not to redefine ourselves as a band, but to evolve our working practice into an ongoing process of escaping definition altogether.

The practice-based research contained in this study, the public voice through which 65daysofstatic talked about and chose to present these ideas and music into the world, and this written commentary, are all accounts from the same author that offer different, often contradictory explanations of the same material and compositional processes, and through a dialectical process they can coexist. In theoretical terms, the interstitial space between academia and popular music this study inhabits is suggestive of Adorno's negative dialectics rather than the Hegelian mode, because the space is not exactly a synthesis of the contradictions drawn from both sides. For Adorno “[negative] dialectics is the consistent sense of nonidentity” (1973), whereas the act of formalising contradictions involves a reductive act of identification because “objects do not go into their concepts without leaving a remainder” (1973). In other words, the true nature of this space is not revealed or synthesised through the contradictions between the academic and popular music domains, rather it is that the contradictions can only reveal that at least a space or, indeed, the potential for many spaces, does exist. This is where *Decomposition Theory* fits. A critical framework of composition that is aware of its own constraints and prospects. A tool that does not, by itself, invent a new future for music, but can reveal that the potential for such futures do exist.

Appendices

Appendix I: 65daysofstatic Discography

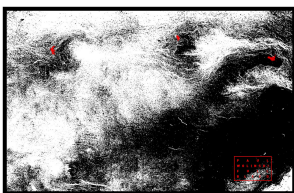
ALBUMS

2004 - The Fall of Math (Monotreme Records)
2005 - One Time For All Time (Monotreme Records)
2007 - The Destruction of Small Ideas (Monotreme Records)
2010 - We Were Exploding Anyway (Hassle Records)
2011 - Silent Running (Dustpunk Records)
2013 - Wild Light (Superball Records)
2016 - No Man's Sky: Music for an Infinite Universe (Laced Records)
2019 - replicr, 2019 (Superball Records)

Singles, E.Ps & Live Records

2003 - stumble.stop.repeat (Dustpunk Records)
2005 - Hole (Monotreme Records)
2006 - Radio Protector (Monotreme Records)
2007 - Don't Go Down to Sorrow (Monotreme Records)
2008 - The Distant and Mechanised Glow of Eastern European Dance Parties (Monotreme Records)
2009 - Escape From New York (Live album) (Monotreme Records)
2010 - Heavy Sky (Hassle Records)
2010 - Crash Tactics (Hassle Records)
2010 - Come to Me/Weak 4 (Hassle Records)
2013 - Prisms (Superball Records)
2013 - Taipei (Superball Records)
2016 - Supermoon (Laced Records)

Appendix II: Full Bleed



Paul Wolinski - Full Bleed. Released on Sacred Tapes, 2014.

Available in local files.

Listen on Bandcamp: [link](#)

Appendix III: YrCodelsBadAndYouShouldFeelBad

The original full code published by 65daysofstatic is available as a PDF in the local files in the 'Appendix III' folder. Visit <https://gibber.cc>, follow links and instructions to listen/manipulate within a web browser.

Appendix IV: Fully Automated Luxury Composition

This journal article is included as a PDF in the local files in the 'Appendix IV' folder. The accompanying audio, *Fully Automated Luxury Composition*, can be found in the audio folders for Chapter 3 - Phase Two. To access the article directly from the journal please visit: https://iaspmjournal.net/index.php/IASPM_Journal/article/view/814.

Appendix V: Zombie Coding

This unpublished journal article is included as a PDF in the local files in the 'Appendix V' folder. The accompanying video, *Zombie Coding*, can be found in the local folder '*Chapter 3 - Phase Two*'.

Appendix VI: Behaviour Tree Notes

These unedited notes are included as extra autoethnographic material as a PDF in the local files in the 'Appendix VI' folder. (See section 3.4.2 - Leaves).

Appendix VII: Future Music Interview



This magazine article is included as a PDF in the local files in the folder 'Appendix VII'.

Bibliography

65daysofstatic. (2004). *The Fall of Math* [CD, Vinyl, Digital]. London: Monotreme Records.

65daysofstatic. (2010). *Debutante*. On 'We Were Exploding Anyway' [CD, Vinyl, Digital]. London: Hassle Records.

65daysofstatic (2010). *Inside* performed by Jean Abreu Dance Company &

65daysofstatic. Edinburgh Festival, Edinburgh. [6-12 August, 2010].

65daysofstatic. (2011). *Silent Running* [Vinyl, Digital]. Sheffield: Dustpunk Records.

65daysofstatic & Rutherford, A (2012). *Every Space Shuttle Ever Launched, in Order* performed by Adam Rutherford & 65daysofstatic. Hammersmith Apollo, London. [21 December, 2012].

65daysofstatic. (2013). *Wild Light* [CD, Vinyl Digital]. Dortmund: Superball Records.

65daysofstatic. (2013). *Sleepwalk City* [Sound Installation] Exhibited at the Millenium Galleries, Sheffield.

65daysofstatic. (2013). Prisms [Video file]. Retrieved from <https://vimeo.com/75299268>

65daysofstatic. (2014). OSCILLATOR [Video file]. Retrieved from <https://vimeo.com/97882060>

Bibliography

65daysofstatic. (2014). Post on the 65daysofstatic Facebook page. Retrieved from <https://www.facebook.com/65propaganda/photos/a.10150449957105676.425765.59496880675/10152607477075676/?type=1>

65daysofstatic. (2014). *Fugue State [Installation for E-bow and Guitar]*. Exhibited at The Great Gatsby, Sheffield. [31 August 2014].

65daysofstatic (2015). *DJ Set*. V&A Museum, London. [28 August, 2015].

65daysofstatic (2016). *No Man's Sky Live*. Sonar Digital Festival, Barcelona. [16 June, 2016].

65daysofstatic (2016). *Live at A2*. A2, St. Petersburg. [28 March, 2015].

65daysofstatic. (2016). *Supermoon*. On 'No Man's Sky: Music for an Infinite Universe' [CD, Vinyl, Digital]. London: Laced Records.

65daysofstatic. (2016). *Red Parallax*. On 'No Man's Sky: Music for an Infinite Universe' [CD, Vinyl Digital]. London: Laced Records.

65daysofstatic. (2016). 'No Man's Sky: Music for an Infinite Universe' [CD, Vinyl Digital]. London: Laced Records.

65daysofstatic. (2017). Post on the 65daysofstatic Facebook page. Retrieved from https://www.facebook.com/65propaganda/posts/10156069967965676?__tn__=K-R

Bibliography

65daysofstatic. (2019). *Replicr, 2019* [CD, Vinyl, Digital]. Dortmund: Superball Records.

65daysofstatic. (2019). *Trackerplatz*, [Digital]. Dortmund: Superball Records.

65daysofstatic. (2019). *Five Waves*. [Digital]. Dortmund: Superball Records.

65daysofstatic. (2019). *A Year of Wreckage*. [Digital]. Sheffield: Dustpunk Records.

65daysofstatic. (2019). *Kazimir* [Digital]. Sheffield: Dustpunk Records.

Adkins, M. (2014). *A Manifesto of Nodalism*. Journal of Science and Technology of the Arts, 6(1), 51-60.

Adorno, T. W., & Simpson, G. (1941). *On Popular Music*. Zeitschrift für Sozialforschung, 9(1), 17-48.

Adorno, T. (1973). *Negative Dialectics*. UK: Seabury Press.

Adorno, T. W. (2005). *The Culture Industry: Selected Essays on Mass Culture*. Routledge.

Algomech. (2017). *65daysofstatic Concert/Installation*. Retrieved from <https://algomech.com/2017/events/65dos/>

Algorave (2015). Performed by Various Artists. Leftbank, Leeds. [13 July, 2015].

Bibliography

Alpern, A. (1995). *Techniques for Algorithmic Composition of Music*. Retrieved from <http://hamp.hampshire.edu/adaF92/algocomp/algocomp>, 95, 120.

...And You Will Know Us By the Trail of Dead. (1999) *Madonna*. [CD, Vinyl]. Merge Records.

Aphex Twin. (1997). *Bucephalus Bouncing Ball*. On 'Come to Daddy' [CD, Vinyl, Digital]. London: Warp.

Armitage, J. L. (2018). *Spaces to Fail in: Negotiating Gender, Community and Technology in Algorave*. *Dancecult: Journal of Electronic Dance Music Culture*, 10(1).

At the Drive In. (2000). *Relationship of Command*. [CD, Vinyl]. Grand Royal.

Attali, J. (1985). *Noise: The Political Economy of Music* (Vol. 16). Manchester University Press.

Autechre. (2001). *Vi Scose Poise*. On 'Confield' [CD, Vinyl, Digital]. London: Warp.

Barry, R. (2017). *The Music of the Future*. Duncan Baird Publishers.

Bell, R. (2013). *Renick Bell's YouTube Channel*. Retrieved from <https://www.youtube.com/user/renickbell>

Bjorklund, E. (2003). *A Metric for Measuring the Evenness of Timing System Rep-rate Patterns*. SNS ASD Tech Note, SNS-NOTE-CNTRL-100.

Bibliography

Bougaïeff, N. (2013). *An Approach to Composition Based on a Minimal Techno Case Study* (Doctoral dissertation, University of Huddersfield).

Bourriaud, N. (2009). *Altermodern Manifesto: Postmodernism is dead*. Supplanting the Postmodern, An Anthology of Writings on the Arts and Culture of the Early 21st Century. Bloomsbury Publishing USA.

Burland, K., & McLean, A. (2016). *Understanding Live Coding Events*. International Journal of Performance Arts and Digital Media, 12(2), 139-151.

Byrne, D. (2012). *How Music Works*. United States: McSweeney's.

Candy, L. (2006). *Practice Based Research: A guide*. CCS Report, 1, 1-19.

Collins, N., McLean, A., Rohrhuber, J., & Ward, A. (2004). *Live Coding in Laptop Performance*. Organised Sound, 8(03), 1–11. <http://doi.org/10.1017/S135577180300030X>

Collins, K. (2008). *Game Sound: An introduction to the history, theory, and practice of video game music and sound design* (pp. 1–213).

Cox, G., McLean, A., & Ward, A. (2001). *The Aesthetics of Generative Code*. In Proc. of Generative Art.

D'errico, M. (2015). *Worlds of Sound: Indie games, proceduralism, and the aesthetics of emergence*. music, sound, and the moving image, 9(2), 191-206.

Bibliography

Dean, R. T., & McLean, A. (2018). *The Oxford Handbook of Algorithmic Music*. New York: Oxford University Press.

Diver, M. (2015). *No Man's Sky Sounds Amazing, Literally*. Retrieved from https://www.vice.com/en_uk/article/no-mans-sky-sounds-amazing-literally-514

Dodge, C., & Jerse, T. A. (1997). *Computer Music: Synthesis, composition and performance*. Macmillan Library Reference.

Ellis, C., Adams, T. E., & Bochner, A. P. (2011). *Autoethnography: An overview*. *Historical Social Research/Historische Sozialforschung*, 273-290.

Engels, F. (1990). *Ludwig Feuerbach and the End of Classical German Philosophy*. In vol. 26 of Karl Marx, Frederick Engels: *Collected Works*, 353-98.

Eno, B., Bryars, G., & Pachelbel, J. (1975). *Discreet music*.

Eno, B. (2017) *Reflection* [Computer software]. Retrieved from <https://itunes.apple.com/gb/app/brian-eno-reflection/id1180524479?mt=8>

Eno, B. Pitchfork. (2017). *A Conversation With Brian Eno About Ambient Music*. Retrieved from <https://pitchfork.com/features/interview/10023-a-conversation-with-brian-eno-about-ambient-music/>

Eshelman, R. (2000). *Performatism, or the End of Postmodernism*. *Anthropoetics*, 6(2), 2001. Chicago.

Bibliography

Facebook (2017). *65propaganda*. Retrieved from <https://www.facebook.com/65propaganda/posts/10155869394300676>.

Finer, J. (1999). *Longplayer*, [Sound installation]. Exhibited at Trinity Buoy Wharf, London 31st December 1999 - 31st December 2999.

Fisher, M. (2011). *Capitalist Realism: Is there no alternative?* London: John Hunt Publishing.

Fisher, M. (2013). *Ghosts of My Life: Writings on depression, hauntology and lost futures*. London: John Hunt Publishing.

Forth, J., Wiggins, G. A., & McLean, A. (2010). *Unifying Conceptual Spaces: Concept formation in musical creative systems*. *Minds and Machines*, 20(4), 503-532.

Gibber. (2014). *Gibber*. Retrieved from <https://gibber.mat.ucsb.edu>

Hainge, G. (2013). *Noise Matters: Towards an ontology of noise*. Bloomsbury Publishing USA.

Harper, A. (2011). *Infinite Music: Imagining the next millennium of human music-making*. John Hunt Publishing.

Harper, A. Verso. (2015). *On Music and Folk Politics*. Retrieved from <https://www.versobooks.com/blogs/2367-adam-harper-on-music-and-folk-politics>

Bibliography

Hart, I. (2014). *Meaningful Play: Performativity, Interactivity and Semiotics in Video Game Music*. *Musicology Australia*, 36(2), 273-290.

Hawthorne, K. (2019, May 2). *Holly Herndon: the musician who birthed an AI baby*. The Guardian. Retrieved from <https://www.theguardian.com/music/2019/may/02/holly-herndon-on-her-musical-baby-spawn-i-wanted-to-find-a-new-sound>

Hello Games. (2016). *No Man's Sky* [Computer software]. London, Sony.

Ihde, D. (2007). *Listening and Voice* (2nd ed.). Albany: State University of New York Press.

Intermorphic. (2019). *Generative Music & Brian Eno's GM1 with SSEYO Koan Software*. Retrieved from <https://intermorphic.com/sseyo/koan/generativemusic1/>

Jameson, F. (1991). *Postmodernism, or, the Cultural Logic of Late Capitalism*. Duke University Press.

Jameson, F. (2005). *Archeologies of the Future*. New York: Verso.

Jameson, F. (2003). *The End of Temporality*. *Critical Inquiry*, 29(4), 695-718.

Jameson, F. (2009). *Valences of the Dialectic* (p. 303313). London: Verso.

Jayemanne, D. (2017). *Performativity in Art, Literature, and Videogames*. Springer.

Bibliography

Kane, B. (2007). *L'Objet Sonore Maintenant: Pierre Schaeffer, sound objects and the phenomenological reduction*. *Organised sound*, 12(1), 15-24.

Khatchadourian, R. (2015, May 18). *World Without End*. *The New Yorker*.

Kirn, P. (2011). *Circles and Euclidean Rhythms: Off the Grid, a Few Music Makers That Go Round and Round*. Retrieved from <http://CDm.link/2011/03/circles-and-euclidian-rhythms-off-the-grid-a-few-music-makers-that-go-round-and-round/>.

Klumpenhouwer, H. (2013). *Commodity-form, Disavowal, and Practices of Music Theory*. In R. Qureshi (Ed.) *Music and Marx*

Kundera, M. (2003). *The Art of the Novel*. London: Harper Collins.

Laing, R. (2016). *How 65daysofstatic Built the Soundtrack to No Man's Sky's Infinite Universe*. Retrieved from <https://www.musicradar.com/news/guitars/how-65daysofstatic-built-the-soundtrack-to-no-mans-skys-infinite-universe-641184>.

Long Now Foundation, 2003, Brian Eno Interviewed by D. Hillis, *Seminars About Long-term Thinking*, 14 November. <http://longnow.org/seminars/02003/nov/14/the-long-now/>.

Ludomusicology. (2019). Retrieved from <http://www.ludomusicology.org>.

Magnusson, T. (2014). *Herding Cats: Observing Live Coding in the Wild*, 1–9. <http://doi.org/10.1162/COMJ>

Bibliography

Magnusson, T., & Sicchio, K. (2016). *Writing With Shaky Hands*. International Journal of Performance Arts and Digital Media, 12(2), 99–101. <http://doi.org/10.1080/14794713.2016.1234689>

Magnusson, T. (2011). *Algorithms as Scores: Coding Live Music*. Leonardo Music Journal 21, 19-23. <https://www.muse.jhu.edu/article/457976>.

Manuel, P. (2013). *Modernity and Musical Structure: Neo-Marxist Perspectives on Song Form and Its Successors*. In R. Qureshi (Ed.) *Music and Marx*

Marx, K., (1859). *A Contribution to the Critique of Political Economy*. Moscow: Progress Publishers: Notes by R. Rojas, 1977.

McClean, A. (2006). *Alex McClean's YouTube Channel*. Retrieved from <https://www.youtube.com/user/yaxu>

Miranda, E., & Williams, D. (2015). *Artificial Intelligence in Organised Sound*. Organised Sound, 20(1), 76-81. doi:10.1017/S1355771814000454

Mixmag. (2017). *Algorave: The Live Coding Movement That Makes Next-Level Electronic Music*. Retrieved from <https://mixmag.net/feature/algorave/>

Montuori, A. (2011). *Beyond Postnormal Times: The future of creativity and the creativity of the future*. Futures, 43(2), 221-227.

Moore, A. W. (2018). *The Infinite*. Routledge.

Bibliography

Native Instruments. (2017, July 5). *Neven on creating Polyrythms with REAKTOR 6 | Native Instruments* [Video file]. Retrieved from <https://youtu.be/WCRWFOSNSvg>

Nawar, T. (2015). Aristotelian finitism. *Synthese*, 192(8), 2345-2360. doi:<http://dx.doi.org.libaccess.hud.ac.uk/10.1007/s11229-015-0827-9>

Now Then. (2017). *Live Reviews*. Retrieved from <http://nowthenmagazine.com/sheffield/issue-117/live-reviews/>

Parkinson, A., & Bell, R. (2015). *Deadmau5, derek bailey, and the laptop instrument-improvisation, composition, and liveness in live coding*. Proceedings of the First International Conference on Live Coding.

Pastebin. (2014). *65dos-yrCodeIsBadAndYouShouldFeelBad*. Retrieved from <https://pastebin.com/GtEXe60y>

Phillips, W. (2014). *A Composer's Guide to Game Music* (pp. 1–285).

Plans, D., & Morelli, D. (2012). *Experience-driven procedural music generation for games*. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(3), 192-198.

Playstation Experience (2014). *65daysofstatic perform A Night Under No Man's Sky*. PSX, Las Vegas. [6 December, 2014]

Bibliography

Playstation. (2014, June 9). *No Man's Sky Gameplay Trailer | E3 2014 | PS4* [Video file].

Retrieved from <https://youtu.be/nLtmEjqzg7M>

Playstation. (2014, December 6). *No Man's Sky - Gameplay Trailer | PS4* [Video file].

Retrieved from <https://youtu.be/nmwG6Sj1Yfg>

Playstation Europe. (2017, March 30). *No Man's Sky | 65daysofstatic Studios Session | PS4* [Video file]. Retrieved from https://youtu.be/_OAKUIRINF0

Pohjannoro, U. (2016). *Capitalising on intuition and reflection: Making sense of a composer's creative process*. *Musicae Scientiae*, 20(2), 207-234.

Polinski (2011). *Labyrinths*. [Recorded by Paul Wolinski] [CD, Vinyl, Digital]. UK: Monotreme Records.

Qureshi, R. B. (2013). *Music and Marx: Ideas, Practice, Politics*. Routledge.

Radiohead. (2000). *Kid A* [CD, Vinyl]. XL Recordings Ltd.

Radiohead. (2015). *Polyfauna* [Computer software]. Retrieved from <https://itunes.apple.com/gb/app/polyfauna/id793263070?mt=8>

Reich, S. (1967). *Piano Phase*. Boosey & Hawkes.

Resident Advisor. (2019, Feb 4). *Algorave Generation* [Video file]. Retrieved from <https://youtu.be/S2EZqikClfY>

Bibliography

Reynolds, S. (2011). *Retromania: Pop culture's addiction to its own past*. Macmillan.

Roberts, C. (2018). *Real time Annotations & Visualizations in Live Coding Performance*.

Splash: LIVE Programming Workshop, Boston, Retrieved from [https://](https://charlieroberts.github.io/annotationsAndVisualizations/)

charlieroberts.github.io/annotationsAndVisualizations/

Roberts, C. (2016). *Code as Information and Code as Spectacle*. International Journal of Performance Arts and Digital Media, 12(2), 201–206. [http://doi.org/](http://doi.org/10.1080/14794713.2016.1227602)

10.1080/14794713.2016.1227602

Rudrum, D., & Stavris, N. (Eds.). (2015). *Supplanting the Postmodern: An Anthology of Writings on the Arts and Culture of the Early 21st Century*. Bloomsbury Publishing USA.

Schaeffer, P. 1966. *Traité des objets musicaux*. Paris: Éditions du Seuil.

Schaeffer, P., Mâche, F. B., Philippet, M., Bayle, F., Ferrari, L., Malec, I., & Parmegiani, B. (1967). *La musique concrète*. Presses universitaires de France.

Schafer, R. M. (1993). *The Soundscape: Our sonic environment and the tuning of the world*. Simon and Schuster.

Scrivener, S. (2002). *The Art Object Does Not Embody a Form of Knowledge*. Working Papers in Art and Design, <http://www.herts.ac.uk/artdes1/research/papers/wpades/vol2/scrivenerfull.html>

Bibliography

Second Wind Ltd. (2011). *Björk: Biophilia* [Mobile application software]. Retrieved from <https://apps.apple.com/gb/app/björk-biophilia/id434122935>

Shrewsbury, J. and Wolinski, P. (2016, September). *In Space, Nobody Can Hear You Score*. A talk at EGX, Birmingham.

Simpson, C Gamasutra. (2014). *Behaviour Trees for AI: How They Work*. Retrieved from https://www.gamasutra.com/blogs/ChrisSimpson/20140717/221339/Behavior_trees_for_AI_How_they_work.php

Sonic Couture [Soniccouture]. (2017, April 4). *Electro-Acoustic Beat Tools 1: Euclidean Beats* [Video file]. Retrieved from <https://youtu.be/nXvcABQMmEU>

Small, C. (2011) *Musicking : The Meanings of Performing and Listening*. United States: Wesleyan University Press

Smalley, D. (1997). *Spectromorphology: explaining sound-shapes*. Organised Sound, 2(2), pp.5–13.

Srnicek, N., & Williams, A. (2015). *Inventing the Future: Postcapitalism and a world without work*. Verso Books.

Spore (Electronic Arts, 2008).

Stokes, M. (2013). *Marx, Money, and Musicians*. In R. B. Qureshi (Ed.) *Music and Marx: Ideas, Practise, Politics*. New York: Routledge: 139-163.

Bibliography

Takahashi, K. (2019). *Keiijiro Takahashi*. Retrieved from <https://www.keiijiro.tokyo>.

Teenage Engineering. (2019). *Videolab*. Retrieved from <https://github.com/teenageengineering/videolab>.

The Game Awards (2016). *65daysofstatic perform Supermoon*. The AXIS, Las Vegas. [5 December, 2014].

Toplap. (2010). *ManifestoDraft*. Retrieved from <https://toplap.org/wiki/ManifestoDraft>

Toussaint, G. T. (2005, July). *The Euclidean Algorithm Generates Traditional Musical Rhythms*. In Proceedings of BRIDGES: Mathematical Connections in Art, Music and Science (pp. 47-56).

Twitter. (2017). Twitter Search. Retrieved from (<https://tinyurl.com/65decompTwitterSearch>).

Tytel, M. (2019). Matt Tytel. Retrieved from <https://tytel.org>.

Tytel, M. (2019). Audio Helm [Unity Asset]. Retrieved from <https://tytel.org/audiohelm/>

VGX Awards (2013). *No Man's Sky Launch Trailer*. Las Vegas. [7 December, 2013].

Webber, C. (2009). *Creating a Virtual Heart*. Retrieved from <http://colinwebber.com>.

Bibliography

Wertheimer, M. (2014). *Music, Thinking, Perceived Motion: The Emergence of Gestalt Theory*. *History of Psychology*, 17(2), pp.131–133.

Wired. (2017). The No Man's Sky Band is Back. Retrieved from <https://www.wired.co.uk/article/65daysofstatic-band-algomech-festival-performance-no-mans-sky-decomposition-theory>

Wolinski, P. (2014). *Wrote Through Ghosts To Get Here: Ghosts Of My Life & Cultural Paralysis*. Retrieved from <http://thequietus.com/articles/15574-paul-wolinski-65daysofstatic-mark-fisher-ghosts-of-my-life>

Wolinski, P. (2014). *Full Bleed*. [Recorded by P. Wolinski]. [Cassette, Digital]. United Kingdom: Sacred Tapes.

Wolinski, P. (2015). *Full Bleed A/V*, performed by Paul Wolinski. Sonic Pattern, Access Space, Sheffield. [19 June, 2015].

Wolinski, P. (2015). *Full Bleed A/V*, performed by Paul Wolinski. Idle Hands, Manchester. [24 July, 2015].

Wolinski, P. (2015). *Full Bleed A/V*, performed by Paul Wolinski. Artsmith Studios, Derby. [8 August, 2015].

Wolinski, P. (2015). *Full Bleed A/V*, performed by Paul Wolinski. Liverpool Psyche Festival, Liverpool. [25 September, 2015].

Bibliography

Wolinski, P. (2015). *Live A/V*, performed by Paul Wolinski. A38, Budapest. [29 January, 2018].

Wolinski, P. (2015, September). *If in Doubt, Make it Loud*. Lecture-recital presented at the (Per)Forming Art Symposium, Leeds.

Wolinski, P. (2017). *ZGEN*, performed by Paul Wolinski. Access Space, Sheffield. [19 May, 2017].

Wolinski, P. (2017). Fully Automated Luxury Composition. *IASPM Journal*, 7(2), 8-15.

Woods, A., & Grant, T. (2003). *Reason in Revolt, Vol. II: Dialectical Philosophy and Modern Science*. Agora Publishing.

Xenakis, I. (1992). *Formalized music: thought and mathematics in composition* (No. 6). Pendragon Press.

Zimmerman, Joel. 2013. "We All Hit Play." <http://deadmau5.tumblr.com/post/25690507284/we-all-hit-play>.

Zmölning, I. M. (2016). *Audience Perception of Code*. *International Journal of Performance Arts and Digital Media*, 12(2), 207–212. <http://doi.org/10.1080/14794713.2016.1227604>